

MACHAZINE

Jaargang 15 - Nummer 1
Oktober 2010

Wiskunde
Informatica
Studievereniging



'Christiaan
Huygens'

A NEW START

FSR

.....
TRANSFORMING THE MATRIX

EJW

.....
APPLIED TO VOTING THEORY

Ehrhart Theory

.....
HISTORISCH MACHAZINE

Machazine
Lustrum

VASTE RUBRIEKEN:

Actueel | Vereniging | Informatica | Wiskunde | Supplementair

Tomorrow, we will be able to make chips faster. Today, you can tell us how.

Deep UV-light
(193 nm)

The race to increase the number of IC switches per square centimeter is not the only race that is underway in the chip world. Manufacturers are also aiming to accelerate chip production. But how do you boost a machine that needs to be accurate to the nanometer?

ASML is now working on chip lithography systems in which a disk of photo-sensitive silicium (the wafer) is illuminated at high speed.

The wafer lies on the so-called wafer stage, which weighs more than 35 kilos. It is passed back and forth under the light, with an extreme acceleration and deceleration of 33 m/s^2 .

33 m/s^2



Chips with 45 nm details can only be made if, between acceleration and deceleration, you illuminate the wafer precisely to the nanometer. One thousand sensors and 800 actuators control and, consequently, illuminate 180 wafers an hour. How much software and how many processes are required to do this? And how do you manage the necessary IT architecture?

Accelerating by 33 m/s^2 poses a challenge in itself. Which motors do you choose? Where do you find amplifiers with 100kW capacity, 120 dB SNR and 10 kHz BW? And that is just the beginning – because the heat itself distorts the accuracy of your system as well...

For engineers who think ahead

Profile: Worldwide market leader in chip lithography systems | Market share: 65% | R&D-budget: EUR 500 million | Opportunities for: Physicists, Chemists, Software Engineers, Electrotechnicians, Mechatronics and mechanical engineers | Discover: ASML.com/careers



ASML



inhoud en colofon

MACHAZINE

is een uitgave van

W.I.S.V. 'Christiaan Huygens'

Hoofredacteur

Dorien de Regt

Redactie

Max de Groot, Kees Boon, Wikash Sewlal,
Michiel van Dam, Friso Abcouwer, Derk-Jan
Karrenbeld, Merel Stout (QQ'er)

Beeldredactie

Bruno Scheele, Peter Pul, Anne van Ee

Redactieadres

Mekelweg 4, 2628 CD Delft

E: machazine@ch.tudelft.nl

T: 015-2782532, F: 015-2782690

Concept en ontwerp

G₂O Kesteren

Drukker

DeltaHage bv

Cover

Collage van MaCHazine historie

Aan dit nummer werkten mee

Bas van Sambeek, Arie Troebel, Prof.dr. C.
Witteveen, Remy Alidarso, Tom Verhoeff, Ankur
Sharma, Pim Stuurman, Frank Tabak, Marijn Goe-
degebure, Sanne Aalbers BSc., Maniek Santokhi,
Michelle Post, Dr. P.G. Kluit, Dr. D.M.J. Tax,
Freena Eijffinger, Tristan Timmermans, Xanvier
Brouwer, Keorb Redno, Daco Harkes, Prof.dr. J.M.
Aarts, Thea Vuik, Frank Tabak, Peter Disselkoen

Algemene Voorwaarden

De MaCHazine-commissie en het bestuur zijn
verantwoordelijk voor de inhoud van het
Machazine, met dien verstande dat de mening
van een schrijver niet (noodzakelijk) de mening
van de redactie of de vereniging weergeeft.
Alle rechten voorbehouden. Niets uit deze
uitgave mag worden verveelvoudigd, opgeslagen
in een geautomatiseerd gegevensbestand of
openbaar gemaakt, in enige vorm of op enige
wijze, hetzij elektronisch, mechanisch, door
fotokopieën, opnamen of enige andere manier,
zonder voorafgaande schriftelijke toestemming
van de uitgever.

Adverteerdersindex

Technolution	Achterkant Kaft
ASML	Binnenkant Kaft
Enrichment Technology	Binnenkant Kaft
Flowtraders	18
Optiver	35



Actueel

Redactioneel - The Pantone Menace	2
Agenda	2
Arie Troebel - Beetje Versleutelen!!	3
Van Bestuur 54 - Remy Alidarso	4
Column - TODO	5
Faculty Student Council - A new start	6
Column international student - Come with me to.....Shanghai	7

Vereniging

Eerstejaarsweeken 2010 - Transforming the Matrix	8
Het EJW door de ogen van nieuwe eerstejaars	10
Bestuurwissel 54	11
EJW-Reünie	11
Docent van het Jaar - Een jaar vol met goeie docenten	13

Informatica

Software-Archivering met Emulatie	14
Ken je studieadviseurs!	16
Gadget - Dagelijkse besommingen	19
TU Delft Dreamteam - DUT Racing	20
Een kennismaking met Microsoft Surface	22
Informaticapuzzel XXIV - Partition	24
Oplossing Informaticapuzzel XXIII - Quadtree codering	25
Pattern Recognition - Hij doet het niet, of toch...?	26
Nuna Serious Racing at the Delft Science Centre	28
BSc Project Technische Informatica - Project Pacman	30

Wiskunde

Reconstructie van de aardkorst met tomografie	32
Ehrhart Theory applied to Voting Theory	34
Wiskundepuzzel LII - Prelude tot de isochronie	36

Supplementair

Alumnus - Peter Disselkoen	38
Historisch Persoon - Gottfried Wilhelm Leibniz	40
Historisch Machazine	42



Bas van Sambeek

Soldaat van Sambeek, ingerukt, Snickers!



'Studenten vinden zichzelf goed in bed.' Zijn dat dezelfde die 30 EC en een 6,0 gemiddeld genoeg vinden :D ?

October

November

Tue 23 Poker Tournament

December

Thu 16 Christmas Dinner



Arie Troebel

Beetje Versleutelen!!

Lieve schaapjes in het mooie Delft,

Het nieuwe studiejaar is weer begonnen en we popelen met zijn allen om weer plaats te nemen in de collegebanken, want zeg nu eens, is er nog iets mooiers te bedenken dan studeren?

Ook hier op Ter-Weksel is het weer een drukte van belang. De nieuwe lichting kan je er zo tussenuit halen. Zij lopen met hun spiksplinternieuwe laptop met het Ter-Wekselse logo erop door de gangen te paraderen. We moeten ze een beetje in de watten leggen toch?

Ik heb ook een heerlijke vakantie gehad. Omgeven door mijn boeken konden mijn dagen niet stuk. Goed, ik was vergeten om naar buiten te gaan tijdens de twee weken dat ik op mijn hotelkamer zat op het paradijselijke eiland Bali. Op de dag van uitchecken deed ik mijn gordijnen pas open. Wat wil je als je Bernard Riemann's (1826-1866) tot nu toe onmogelijk op te lossen probleem, de Riemann-hypothese, hebt opgelost!

De mensen bij de in- en uitcheckbalie van mijn hotel op Bali vonden uiteindelijk dat ik niet zo lekker rook, maar een kniesoor die daar wat van zegt, toch lieve schatjes?

Maar goed ik dwaal even af. De opening van het academisch jaar moest gehouden worden en niemand minder dan G. Flater had dit keer de twijfelachtige eer dit te mogen doen.

Meneer was al een aantal weken druk bezig geweest met iets wat uit zijn onvoorspelbare brein omhoog was gekomen. Hij had zichzelf ook opgesloten op zijn werkkamer, maar ik denk niet dat dat een goed teken was.

Ik hield mijn wetenschappelijke hart vast.

Nadat ik de aanwezigen, waaronder de burgervader en de minister van onderwijs die elk jaar aanwezig is, in ons immens auditorium had verwelkomd, gaf ik de heer Flater het sein dat hij van het Italiaans marmeren podium gebruik kon gaan maken.

En lieve wetenschappers in de dop dat hebben wij geweten!!

Flater stelde zichzelf voor, en ik moet zeggen dat ging foutloos in elkaar over; de voornaam en achternaam kwamen er duidelijk en foutloos uit. Maar toen begon het....

Meneer hield zijn hele, maar dan echt zijn hele betoog in 'Caesarcodes C5' afgewisseld met op het krijtbord, jawel, monosyllabische substitutie. Het ging maar door en door!

Na zestig minuten was de foltering voor de 2.500 aanwezigen voorbij en liep iedereen sufgepraat en tollend op z'n benen, alsof ze 14 uur in een pub onafgebroken hadden gedronken, naar het enige redmiddel wat nu echt van groot belang was. Inderdaad: naar de bar.

Het laatste wat ik hoorde toen ik het bewustzijn verloor was dat Flater tegen mij zei 'Arie hoe vond je het?' 





Remy Alidarso, Voorzitter Van het Bestuur 54

Het is OntvangstWeek in Delft, bij ieder van jullie ongetwijfeld beter bekend als de OWee. Het regent nonstop, en wat kun je dan beter doen dan een stukje schrijven voor het eerste MaCHazine dat dit jaar zijn lustrum viert? Niet dat er niet genoeg ander werk op de plank ligt: er moet namelijk nog vergaderd worden over het een en ander, er moeten nog tentamens geleerd worden, er moet nog gewerkt worden, en daar houdt de lijst niet op. Toch grijp ik het moment om even te ontspannen en tegelijkertijd productief bezig te zijn.

Als aankomend bestuurder ziet de zomer tussen twee collegejaren er anders uit dan normaal. Natuurlijk, je hebt net als altijd wat meer tijd vrij te maken voor leuke dingen: terrasje hier, bioscoopje daar, strandje zus, BBQ zo, etc. Maar er moeten ook serieuze dingen gebeuren voor de werkzaamheden die je het nieuwe collegejaar allemaal gaat uitvoeren. Overdrachtsessies, vergaderingen, bedrijfsbezoeken, kleding regelen en ga zo maar door. Omdat het allemaal nieuw is, wordt dit niet ervaren als vervelend, maar juist als een uitdaging. En ik denk dat dat vaker zou mogen gebeuren. Aan mezelf merk ik het verschil wanneer ik aan een opdracht begin en deze zie als een uitdaging, of wanneer ik de opdracht zie als iets waar niets uitdagends aan is. Opdrachten van categorie één zal ik ook zeker met de nodige creativiteit verrijken, om er nóg wat mooiers van te maken dan anders.

Zie bijvoorbeeld dit MaCHazine. In de inleiding heb ik het al vluchtig genoemd, maar dit is de eerste uitgave van het MaCHazine dat dit jaar zijn derde lustrum viert. En ik kan niet anders zeggen dan dat de kwaliteit, zowel fysiek, als inhoudelijk, er met de jaren alleen maar op vooruit is gegaan. De mensen van de MaCHazine-commissie gaan steeds weer de uitdaging aan om de kwaliteit naar een hoger niveau te tillen. Er wordt telkens gekeken naar hoe het blad verbeterd kan worden. En het doet mij deugd te zien dat veel andere commissies dezelfde insteek hebben om hun activiteit weer op een hoger niveau te brengen.

Over activiteiten gesproken: ook dit jaar zullen er interessante lezingen georganiseerd worden, zullen er bedrijfsexcursies plaatsvinden, zal de verjaardag van onze studievereniging goed gevierd worden tijdens de DIES NATALIS, waarbij uiteraard een nieuw jaarboek gepresenteerd zal worden, zal er later in het jaar een bedrijvenreis plaatsvinden, zal er een aantal ledenlunches georganiseerd worden, en zullen er natuurlijk een aantal pre-lustrum activiteiten plaatsvinden, want bestuursjaar 55 is natuurlijk weer een lustrumjaar. Al met al dus genoeg activiteiten om je weer een jaar lang niet te hoeven vervelen.

Als bestuurder van een studievereniging hoef je je trouwens nooit te vervelen heb ik me laten vertellen, want er is altijd wel iets te doen. Maar ik ben dan natuurlijk ook niet van plan om er een saai jaar van te maken, want zeg nou zelf, een jaar lang een vereniging leiden met 5 mede bestuursgenoten, dat kan toch niet als saai, maar alleen maar als spannend en uitdagend worden ervaren? Ik heb in ieder geval erg veel zin om me samen met deze 5 mensen een jaar lang voor de vereniging in te zetten door onder andere leuke activiteiten neer te zetten. Dat doen we natuurlijk niet alleen, maar samen met jullie, onze leden!

Op het moment van schrijven is het inmiddels alweer twee weken na de OWee, en dat betekent dat de bestuurswissel voor de deur staat. Nu gaat het voor ons pas allemaal echt beginnen. Wanneer jullie dit lezen is dat al lang en breed achter de rug, zijn de eerste collegeweken inmiddels voorbij en zijn jullie als het goed is wel een beetje gewend aan het nieuwe dag en nacht ritme dat je je onmiddellijk aanmeet wanneer je in Delft gaat studeren.

Ik begon dit stukje te schrijven toen de OWee net van start was gegaan en ik heb steeds een kort moment van ontspanning genomen om het stukje uit te breiden om mijn gedachten even ergens anders op te kunnen focussen. Na veel van zulke ontspanningsmomentjes genomen te hebben nader ik de laatste regels van dit stukje, waarin ik nog eens extra wil benadrukken dat het voor mij een plezier was om te schrijven en dat ik er zeker van ben dat jullie na het lezen van dit artikel nog erg veel plezier zullen hebben met het lezen van de overige artikelen. 



Column **TODO**

Prof.dr. C. Witteveen

TODO

Actueel

Faculty Student Council

A new start

Tom Verhoeff

Last year we started posting an update from the Faculty Student Council (FSR) in every MaCHazine. It proved to be useful so the updates will be continued this year. With the start of a new year the newly elected FSR is officially installed so in this update we will introduce the students who will represent all AM & CS and related master students. We will also introduce our high priority points for the next few months.

The council on our faculty consists of two chambers. The chamber ET which has 5 members representing the bachelor EE and masters EE and CE; members of this chamber are usually connected to the study society "ETV". The chamber TWI which has 6 members who represent the bachelors TW and TI and masters AM, CS, MKE and ES. The TWI chamber works closely together with our study society to ensure best results.

During the academic year 2010-2011 the chamber TWI will consist of the following students:

- Sanne Aalbers (AM)
- Linda Crapts (AM)
- Anne van Ee (MKE)
- Wikash Sewlal (TW)
- Tom Verhoeff (TI)
- Xander Zonneveld (TI)

Right now we are still working on the so called "10 point plan" which will list our 10 main points of concern for this year. This faculty will accept this list and use it to make improvements to the points we suggest. In the next MaChazine we will present the final plan, but for now I will introduce some of the points that will definitely be in it.

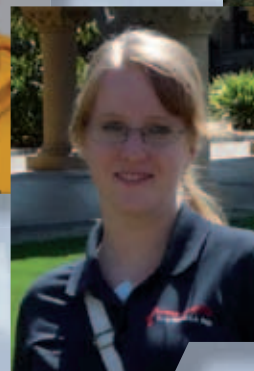
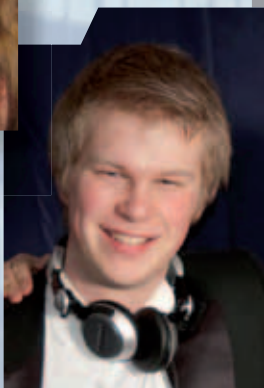
Last year two important new rules have been introduced. Our first year bachelor students were confronted with the "Bindend Studie Advies (BSA)". First year students who receive less than 30 ECTS will not be allowed to continue their current study program. Also, since the first 1st of September pre-master enrollments are not available anymore. This is due to the so- called "harde knip" rule. We will make sure both new rules are seriously evaluated.

Another point that has been on our agenda for some time now is the appearance of our faculty. We are all proud to be a part of our EEMCS faculty, but when a stranger enters our building it is not immediately clear what is happening in our building. We brought this up in meetings with the faculty last year and are now working together to improve the overall appearance of our faculty. If you have any suggestions about improvements on this subject, please let us know!

University-wide the so-called "herijking" has been the most important subject during the last few months. A lot of money has to be saved on a structural basis and this will also have an impact on our faculty. Everyone involved with our university will notice that money has to be saved, but we want to ensure our education does not suffer from the cuts in expenses.

There is one other point we want to share with you. We have seen an increase in cases of theft on our faculty. With people working on laptops everywhere in our faculty, we are an interesting target and something needs to be done about it. Some lockers are available, but they are occupied all the time. Also going to the restroom for a few minutes should be possible without taking all your stuff with you. We will try to improve the situation on our faculty and again suggestions are welcome.

Do you want to respond on this article? Or just want to get in touch with us? Just speak to one of us at the faculty or send us an e-mail at fsr@ewi.tudelft.nl



Left to right: Anne, Sanne, Tom, Xander, Linda and Wikash.



International Student Column

Come with me to.....Shanghai

Ankur Sharma

"Hi Ankur, I am happy to inform you that you have been selected for the 'Come with me to Shanghai' trip. Start packing your bags." You can't imagine how thrilled I was when I got this call. The one week study trip to Shanghai with the Yes! Delft Student Board (www.yesdelft.nl) and 9 other TU Delft students turned out to be as awesome as one could imagine.

I must have flipped through the entire digital entertainment channel during the long Virgin flight from Heathrow but all tiredness vanished as we landed at the giant Shanghai airport. On a Maglev train traveling at 430 km/h, we departed for our destination: a tourist hotel where we met people from all over the world, who had come to see the wonders of Shanghai. However, we were here for much more. Our trip was a study tour with the main intention to answer an important question – How to do business in China. The following week turned out to be an inspiring, motivating and adventurous journey, all packaged in a very well-structured and well-managed deal by the Yes! Delft Student Board.

Our typical day would start at 7 a.m. with a healthy breakfast at the hostel. After that, 13 of us, looking awesome in our suits, would head for our destination. During the week, we met several entrepreneurs, companies and organizations which shaped our perspective on understanding the business scenario in China. After such intellectual meetings and discussions throughout most of the day, the evening was a time to unwind and wear tourist clothing.

One of the main attractions of the trip was the World Expo: a huge canvass of creative, state-of-the art demonstrations of how the cities of the world are adapting to the swiftly changing needs of society. The enormity of the entire event was breathtaking and over the two half days that we visited the Expo, we could hardly cover a small portion of it. With all its glitter, the World Expo clearly showcased the might of this fast-paced city, in a way telling the world that Shanghai and China have arrived and can't be ignored.



Dinners at exotic places, drinks on the 108th floor of the Hilton, shouting our lungs out at the KTV- every evening was a special occasion to experience the Chinese culture. Catching only the minimum amount of sleep that we required, the remaining time was spent in doing as much as we could at this wonderful location. And whenever our spirits were down, the pint-sized Tsingtao beer proved to be a lifesaver. The last day saw most of us shopping at the famous Chinese streets, where a pair of Nike shoes costing 10 euros seemed quite normal!

I would very much like to thank the Yes! Delft Student Board and everyone else at Yes! Delft who arranged a trip for us we'll remember for the rest of our lives. I'm certain that in the coming years, the "Come with me to....." trips will offer more trips like mine to students, so all you entrepreneurs out there: keep your eyes open and apply for the next trip! 📷



Actueel



Eerstejaarsweekend 2010

Transforming the Matrix

Pim Stuurman

Transforming the Matrix: Back to Basic. Oftewel: wat er zou gebeuren als Neo de verkeerde pil nam en samen met Megan Fox een robot-kind zou krijgen dat achterstevoren in Visual kon programmeren. Bent u er nog? Dan neem ik voor het gemak even aan dat u jonger dan 30 bent, of in

ieder geval met een achterwaartse salto over de generatiekloof bent gesprongen. Wees gerust, deze tekst zal verder niet gaan over films, tv-series of programmeertalen. Ik ga u vertellen over het EerstejaarsWeekend 2010!

Zoals elk goed verhaal heeft ook dit een begin, vrijdag 13 augustus om 13:00 uur om exact te zijn. Op dat tijdstip kwamen 100 aankomend eerstejaars

studenten Technische Wiskunde en Technische Informatica aan voor hun eerste ontvangst op onze mooie faculteit. Het EJW (een afkorting, zoals u begrijpt) heeft een traditionele opening, waarbij de deelnemers aan verschillende groeperingen worden voorgesteld. Om dit verslag levendig te houden, verzoek ik u zich in te beelden hoe het is om als eerstejaars een EJW mee te maken. Degene onder u die dit al eens gedaan hebben, kan ik van harte aanbevelen hun EJW-shirt weer eens aan te trekken om de leeservaring te versterken.

“Beste eerstejaarsstudenten”, spreekt een vreemde man in een groen pak u toe, “welkom bij het EerstejaarsWeekend van studievereniging ‘Christiaan Huygens’. Mijn naam is Niels den Ouden en als voorzitter van de WieWie organiseer ik, samen met mijn commissiegenoten, dit weekend voor jullie. Om jullie heen zien jullie een hoop mensen die je nog niet kent. De Wocky!, in nu nog elegant witte badjassen, zal jullie





van eten, drinken en andere consumpties voorzien. De mentoren, enigszins arrogant voor zich uit kijkend, zullen jullie begeleiden gedurende het weekend."

Na een klein lachje hier en daar, vervolgt de man die blijkbaar Niels heet: "Veel belangrijker dan deze groepen zijn echter de mensen in rode shirts. Dit zijn namelijk je aankomende studiegenoten, EJW-maatjes en uiteindelijk je nieuwe vrienden." U kijkt ietwat onwennig om u heen, terwijl na Niels' speech de vicedecaan Ir. M. Sips het woord neemt. Het weekend blijkt dus niet alleen maar uit mensen in rare kleding te bestaan.

Na deze speech neemt Remy, op moment van schrijven nog potentieel voorzitter van CH, het woord. Een korte uitleg over CH en het nieuwe oranje bestuur volgt, waarna u met uw nieuwe vriendjes een rondleiding krijgt door de faculteit. De verschillen tussen een universiteit en uw middelbare school worden al snel duidelijk: er wordt niet door de gangen gerend, de docenten zijn wat ouder, de zalen iets groter en de gymzaal is vervangen door een bar.

Wanneer iedereen zich buiten verzamelt, kan het eerste stukje onderwijs van uw studietijd beginnen. Lineaire Algebra 0 begint met een uitleg over een matrix. Na enige onwennigheid vormt u al snel matrices van allerlei verschillende afmetingen, samen met uw jaargenoten. Het inverteren blijkt hier en daar nog wat lastig, maar de basis is er.

Nadat Niels weer een half uur van uw leven heeft verspild met zijn volgende blaafverhaal, kunnen eindelijk de bussen vertrekken. Het nieuwe bestuur zorgt voor het vermaak met interessante vraagstukken als "hoe heet je?" en "hou je van hobby's?", waardoor de uren injectief op minuten worden afgebeeld.

Op de locatie aangekomen begint uw tweede college: Zingen 1. Na een paar oefenliedjes als "ééééén nuuuul, één nul één nul één nuuuul" en "Integraal, wie kent hem niet?", wordt het tijd voor het echte werk. Het WIM (Wiskunde, Informatica en misschien kan MKT eigenlijk volgend jaar wel weg)-lied wordt geprobeerd, gefaald en nog vele malen herhaald.

Na wat Wocky!-wraps gegeten te hebben kan het feest dan eindelijk losbarsten. Uw eerste feest als student! Verschillen met de middelbare schoolfeesten worden ook nu weer snel zichtbaar: er is meer bier, minder vrouwen en de docenten Zingen 1 en Lineaire Algebra 0 lijken sneller dronken te zijn dan

u zelf. Dat had die stoffige wiskundeleraar op de middelbare school toch nooit?

Al snel denkt u er niet meer over na en komt u tot de conclusie dat uw jaargenoten toch echt wel toffe gasten zijn. DJ Evers en Krikke draaien er lekker op los en u draait lekker mee. Als het draaien op een gegeven moment niet meer stopt terwijl u stil staat, besluit u toch maar eens een bed te gaan zoeken.

De volgende ochtend blijkt dat er ook dingen zijn die niet veranderen als men gaat studeren. De zon staat nog steeds eerder op dan u en katers veroorzaken ook nog net zo veel hoofdpijn als voorheen. Tot overmaat van ramp heeft een of andere idioot ook nog bedacht dat ochtendgymnastiek een goed idee was.

De rest van de dag wordt gespendeerd aan meer sport, die toch stiekem best leuk is. Vooral het zeepvoetbal levert genoeg hilariteit op, waarbij u zich verbaast over het feit dat niemand geblesseerd raakt. Daarna loopt u een korte vossenjacht, waarbij u CH iets beter leert kennen. Eigenlijk had u gewild dat deze tocht nog iets langer zou zijn, maar helaas staat het volgende programma-onderdeel al weer klaar: de Bonte Avond!

Tijdens de openingsact blijkt al snel dat niet alle WieWie'ers Zingen 1 gevolgd hebben. Gelukkig mogen uw jaargenoten eerst nog hun kunsten vertonen als dansers, dwergen en (in het geval van sommige bestuursleden) Delftsche dames. Als ook uw eigen act soepel verlopen is, kan de Bonte Avond afgesloten worden en de laatste avond beginnen.

Vreemd genoeg loopt die net als de avond ervoor. Er wordt gesjanst, gedanst en gedronken en juist degene die vanochtend nog riep dat hij op tijd naar bed ging staat tot zonsopgang los te gaan. De volgende ochtend staat u op en na wat laatste activiteiten gaat u tevreden in de bus zitten. Op de weg terug naar huis kijkt u naar de slapende gezichten om u heen en beseft u zich dat die rare groene idioot toch gelijk had. Hier zitten uw studiegenoten, EJW maatjes en nieuwe vrienden.

Pim Stuurman

P.S. U mag uw eerstejaarsshirt nu weer uittrekken. 





Het EJW

Michelle Post

De eerste herinnering die in me opkomt als ik aan het EJW denk, is

het feit dat ik me verbaasde over hoeveel meisjes er waren.

Ik kwam bij EWI aan in de verwachting alleen maar mannen aan te treffen, maar al bij het dumpen van mijn spullen raakte ik aan de praat met een meisje dat ook wiskunde ging studeren. Toch waren de mannen sterk in de meerderheid en dat was ook wel te merken tijdens de avonden; pas na veel bier kwam de boel wat op gang, omdat de jongens over het algemeen iets te verlegen waren om zonder drank los te gaan.

Daarnaast was het EJW een kleurrijk geheel; het toen nog aankomend bestuur in het oranje, de WIEWIE in felgroene pakken, de mentoren in het blauw en de (toen nog) aankomend eerstejaars [lees: nullen] in het rood. Als nullen kwamen we tijdens het EJW in aanraking met begrippen als integreren en matrices. Belangrijk was het afgeven van je mobiel aan het begin van het weekend om het integreerproces soepel te laten verlopen. Niemand was nog bezig met het constant kijken op zijn mobiel, maar met het integreren met jaargenoten. Toch verliep het integreren overdag niet altijd even soepel met iedereen, maar dat kan ook niet anders als sommigen het maken van wiskundegrappen en het spelen van games als hoogtepunt van hun leven beschouwen. Maar tijdens de avonden, na wat bier, kon je met iedereen prima praten.

Op zaterdagmiddag was het tijd voor de vossenjacht. Ik zat in een groepje dat bestond uit ongeveer vijftien mensen, waarvan twaalf vrouwelijk. Er bestaan verschillende vooroordelen over vrouwen, bijvoorbeeld dat vrouwen het richtingsgevoel van een kip hebben. In mijn geval klopt dat wel een beetje, maar wij als groep hebben het tegendeel bewezen: van alle elf groepen is onze groep als enige niet verdwaald. Sommige groepen waren zelfs zo verdwaald dat ze van de verkeerde kant het terrein op waren gekomen en dat ze, toen wij het terrein betraden, verbaasd uitriepen: 'Waar komen zij vandaan?'.

Tenslotte hebben we kunnen genieten van muziek, gezelligheid en een bonte avond, waar mijn groepje een nieuwe kledinglijn met vuilniszakmode heeft gepresenteerd. Daarnaast natuurlijk veel bier en cola, regen en zon, het instuderen van het programmeer- en wiskundecouplet en niet te vergeten de busreis, waar ik schoenen aan namen en hobby's heb kunnen koppelen. Ik kijk met veel plezier terug op het EJW en ik denk dat ik er volgend jaar weer bij ben, maar dan als mentor.

10001010110010101011001010101100101010110010101011
111235813211123581321112358132111235813211123581321
101
901234267890123426789012342678901234267890123426789
011010101001101010100110101010011010101001101010100

Jaargang 15 • Nummer 1 • Oktober 2010

Het begon allemaal ergens in augustus. Een volle zaal met nullen

begroette me. Licht geïntimideerd (ik zal niet de enige zijn geweest)

liep ik naar binnen.

Mijn eerste indruk? Uiterst kleurrijk. Mannen in dikke witte badjassen, gozers in groene pakken en CH-leden in oranje kledij. Daar bleef het niet bij. Een geel kuikentje schreeuwde boven iedereen uit. Vergezeld door een roze panter trokken ze alle aandacht naar zich toe.

Niet veel later werden we begroet door het aankomend bestuur van Christiaan Huygens, bestuur 54. Na een kort praatje werden we naar buiten geloodst. Het was namelijk tijd om in een matrix te gaan staan. Er is een voorzichtige rivaliteit tussen de faculteiten grenzend aan de Mekelweg. Het is dan ook niet vreemd dat er leuke liederen zijn verzonden om die rivaliteit in stand te houden. Hoe kijken de studenten van EWI naar anderen faculteiten? Eerst alles op een rijtje. TB is een cursus, Civiel heeft een voorliefde voor beton, IO is knutselen op universitair niveau en Bouwko's daar beginnen we niet over. Staan in zo'n matrix creëert dus een mooie gelegenheid om deze liedjes te zingen en dat deden we dan ook.

Inmiddels stonden de bussen klaar en iedereen stapte in. Zelfs in de bus werden we bezig gehouden. Het thema: integreren, elkaar leren kennen. Iedereen moest om de beurt naar voren en vertellen wie hij of zij is. Al gauw bleek dat de informatici in de meerderheid waren.

Eenmaal aangekomen en geïnstalleerd kon de pret beginnen. Nachtelijke boswandelingen, sappige voetbalwedstrijden en trefbal. Zomaar een greep uit het programma. Het meest interessante was de trektocht, even geen WieWie aan je kop. Dat groen deed op een gegeven moment toch een beetje pijn aan je ogen.

De voortreffelijke koks van de Wocky zorgden dit weekend voor prima maaltijden. Met Joey aan het roer (lees: roeren in de pannen) kan het ook niet misgaan. Laatst vroeg ik hem nog waarom hij die kekke zonnebril ophad. Waarop hij antwoordde: "Om die wallen te verbergen natuurlijk."

Een pluspunt was dat het bier goedkoop was, ook al was het Grolsch. Na een paar glaasjes vind je het nog lekker ook. De avonden waren daardoor zeer vermakelijk. Zeker de bonte avond, waar iedereen zich een beetje voor lul zette om het publiek te vermaken.

De terugreis herinner ik me weinig van. Ik denk dat velen er zich weinig van herinneren. Iedereen sliep zijn roes uit.

Al met al een briljant weekend!



Briljant EJW!

Maniek Santokhi



Bestuurswissel 54

Marijn Goedegebure

De bestuurswissel van 54 is er een die me nog lang zal heugen. Dit

omdat ik was gevraagd als boekbewaker. Nu was dit concept mij totaal

onduidelijk toen ik werd gevraagd, maar na wat uitleg en het kopen van

een brasjasje was ik er helemaal klaar voor.

Voor de mensen die niet weten wat boekbewaking inhoudt, een korte uitleg: bij de borrel is er een receptieboek aanwezig. Daar mogen besturen in schrijven, maar dit boek willen ze tegelijk ook hebben. Het is dus aan de boekbewakers om te voorkomen dat de andere besturen er met het receptieboek vandoor gaan. Hiervoor zijn door ons meerdere maatregelen genomen, maar daarover later meer.

De borrel begon heel rustig, met het nieuwe bestuur dat werd gefeliciteerd door vrienden en familie. Er waren nog geen andere besturen aanwezig, dus in die tijd hadden wij (de boekbewakers) even de tijd om het plan van aanpak te

bespreken. Zo zou één van ons op een helft van het boek gaan zitten, zodat het moeilijk was om zomaar het boek weg te grissen. Ook gingen we met de rest van de boekbewakers om de tafel heen staan, zodat er maar telkens één iemand bij het boek in de buurt was. De strategie was bepaald en nu was het wachten op de besturen. Eerst mochten de boekbewakers het bestuur feliciteren en het boek aanbieden. We kregen het ook meteen weer terug; we moesten het immers nog bewaken...

Niet lang daarna begonnen de eerste besturen binnen te druppelen. Er waren rustige besturen bij en natuurlijk een aantal prominente besturen. De rustige besturen waren niet zo moeilijk, alleen ontstond er al snel een rij omdat iedereen een kantje vol wil schrijven met zijn of haar goede verhaal. Dit vonden de prominentere besturen natuurlijk niet leuk en er werd dus een hoop geduwd en getrokken om zo snel mogelijk aan de beurt te zijn. Twee keer kwam er een serieuze aanval op het receptieboek. Beide keren probeerden ze door de tafel weg te trekken het boek onder degene die er op zat weg te krijgen. Bij de laatste keer is de tafel gesneuveld en het boek afgevoerd, het was immers al half 7. Het boek was eindelijk veilig!

Ons bestuur was ondertussen weg en het einde van de borrel was nabij. De bar werd dicht gedaan en afgevoerd. De laatste discussies werden met het nodige brassen afgerond en langzamerhand ging het ene na het andere bestuur weg. De rust keerde terug en het was tijd voor het avondeten. 

EJW-Reünie

Merel Stout

Het jaar was nog maar net goed en wel begonnen en de eerste activiteit

stond alweer op de agenda, de EJW Reünie/Commissieinteresseborrel.

Via de trap de /Pub ingelopen, liep je meteen de piraten van de Facie tegen het lijf, die met gepaste trots de Facie-boekjes aan het signeren waren. Zo kreeg iedereen niet alleen een mooi smoelenboek waarin alle EJW-aanwezigen terug te vinden zijn, maar werd er ook nog eens een persoonlijk verhaal voor je in geschreven.

Op weg naar de bar was het ook één groot feest van herkenning, de WIEWIE was uiteraard weer van de partij, net als de Wocky! die als vertrouwd achter de bar de biertjes tapte. Maar de belangrijkste groep was natuurlijk ook de best vertegenwoordigde: de sjaars waren in grote getale naar de /Pub gekomen om gezamenlijk mooie herinneringen aan het Eerstejaarsweekend op te halen.



Na enige tijd lekker geborrel te hebben, werd het tijd voor het echte werk. Te beginnen met presentaties van de twee eerstejaarscommissies: de AKCie en de Sjaarcie. Friso Abcouwer en Karens Grigorjancs legden in het kort uit welke CHave activiteiten ze het afgelopen jaar allemaal hebben georganiseerd en waarom het nou zo leuk is om een commissie te doen bij CH.

Daarna brak het moment aan waar iedereen reikhalzend naar uitgekeken had, de première van de enige echte EJW-film 2010, gemaakt door Tristan Timmermans en Aad van Buuren. De film gaf weer een uitermate goed beeld van het gehele Eerstejaarsweekend, van de binnenkomst op EWI, tot het zingen in de matrix en de vele acts op de Bonte Avond; alles was terug te vinden in de video. Ook de nodige gênante momentjes konden natuurlijk niet weg blijven in dit mooie beeldverslag.

Al met al had iedereen weer het echte EJW-gevoel te pakken gekregen en kunnen we terug kijken op een geslaagde eerste activiteit! 





10 DAYS
3 CITIES
1 GOAL

Business Tour

MAY 2011

Interested?

<http://businesstour.ch.tudelft.nl/>



Docent van het Jaar 2009/2010

Sanne Aalbers, BSc. & Anne van Ee, BSc.

Woensdag 26 mei vond de jaarlijkse Algemene Docenten en Studenten

Lounge en de Docent van het Jaar verkiezing plaats.

Dit jaar werd dezelfde stemmethode gebruikt als in 2008/2009. Studenten kregen een lijst met alle docenten van hun jaar uitgereikt en konden daarop elke docent (waarvan ze dit jaar college van gevolgd hadden) een cijfer geven. Hierna werden de gemiddelde cijfers van de docenten berekend waaruit een top drie volgde. Wat wel anders was van voorgaande jaren, is dat ook master-docenten een kans kregen om mee te dingen naar de titel Docent van het Jaar.

Op de derde plaats is Hans van der Weide geëindigd bij Wiskunde. Je leert hem pas kennen als docent in de master Applied Mathematics, tenzij je de minor Finance hebt gevolgd. Deze minor heeft onder andere hij opgezet en is dit jaar met meer dan 50 studenten een groot succes. Daarnaast geeft hij meerdere vakken in de master en wordt als docent dus erg goed beoordeeld.

Op de tweede plaats staat Ben de Pagter. Naast het feit dat hij het tweede semester erg druk had als plaatsvervangend decaan, was dit ook de eerste keer dat hij het vak Fourieranalyse op zich nam. Dit derdejaarsvak wordt altijd als ontzettend moeilijk beschouwd en staat dan ook bekend als het grootste struikelvak uit de bachelor. De studenten waren echter enorm te spreken over zijn colleges en dit is ook te zien aan de goede resultaten die de studenten hebben gehaald voor dit vak.

Op de eerste plaats is Martin van Gijzen geëindigd als beste wiskundedocent van 2009/2010. Deze docent kom je tegen in je tweede jaar bij het vak numerieke methoden. Studenten zijn altijd erg over hem te spreken en zijn humor wordt altijd enorm gewaardeerd. Naast het feit dat zijn vak vrijwel door de meeste studenten als leuk wordt beschouwd, maken zijn goede colleges het nog beter.

De top drie van Technische Informatica bestond uit Geert-Jan Houben op de derde plaats, Tomas Klos als tweede en de Docent van het Jaar beker was voor Koen Langendoen.

Geert-Jan Houben geeft alleen college aan masterstudenten en dit was pas het tweede jaar dat hij bij onze faculteit college geeft. Zijn enthousiasme voor zijn vakgebied werd erg goed gewaardeerd door de College Responsiegroep Informatica 4.

Voor de tweede keer op een rij heeft Tomas Klos de tweede plaats weten te bemachtigen. Zijn interactieve colleges met clickers zijn duidelijk een hit bij de eerstejaarsstudenten. Hij legt helder en duidelijk uit en dat is heel prettig bij het vak Fundamentele informatica 1: Logica. De meeste studenten hebben met dit vak namelijk veel moeite dus is het erg knap van hem dat studenten alsnog zo enthousiast over hem zijn.

Compiler Construction wordt al jaren als een van de beste Computer Science vakken gezien en dit is vooral dankzij Koen Langendoen. Met de titel Docent van het Jaar van Informatica is het een mooie afsluiting van zijn vak. Wat sommige studenten misschien niet weten is dat Koen zich heel erg actief inzet tegen fraude. Wat iedereen wel weet is dat er biertjes te verdienen zijn als beloning voor het vinden van fouten in zijn slides. Uiteraard worden deze niet tijdens het college genuttigd!

Vanuit elke faculteit wordt een docent voorgedragen die meedingt naar de TU brede Docent van het Jaar prijs. Naast de eer levert dit ook 10.000 euro op! Door dit jaar nog een Evasys enquête uit te zetten onder de studenten, die vakken van de genomineerde docenten hebben gevolgd, wilde wij een goed beeld krijgen van deze docenten. Deze Evasys enquête was gebaseerd op de volgende zeven punten:

De docent...

1. heeft persoonlijk contact met studenten
2. moedigt studenten aan om elkaar te helpen en samen te werken
3. moedigt aan tot actief deelnemen
4. geeft feedback en laat zien wat ik kan verbeteren
5. benadrukt hoe ik mijn tijd effectief kan besteden
6. spreekt hoge verwachtingen uit over mijn prestaties
7. past zich aan mijn manier van leren

Bij elke uitspraak konden studenten aangeven in hoeverre zij het hiermee eens waren. Deze zeven punten komen uit het artikel 'Seven principles on good teaching and learning in colleges and universities' (Chickering & Gamson, first edition 1987). Deze punten worden nu TU breed gebruikt om docenten te beoordelen. Aan de hand van onze resultaten zal de Directeur Onderwijs, Herman Russchenberg, beslissen welke docent vanuit onze faculteit afgevaardigd wordt. Dit is niet per definitie de docent die op de eerste plaats eindigde bij onze verkiezing. De gehele top drie maakt dus kans om de Docent van het Jaar van EWI te worden!

Wij willen nog een keer alle winnaars feliciteren en willen hun bedanken voor hun uitstekende inzet vorig jaar. Op naar een nieuw collegejaar met nieuwe kansen! 



Vereniging

Software-Archivering met Emulatie

Michiel van Dam & Jeff van Egmond

Veel culturele- en onderzoeksdata wordt tegenwoordig gearchiveerd. Bij zo'n archief is het van belang dat het op lange termijn nog mee gaat, dus dat technologische veranderingen het archief niet ontoegankelijk maken. Voor software en andere dynamische data levert dit speciale problemen op, aangezien het meer afhankelijkheden heeft dan normale gegevens. Voor archivering van data zijn er op het moment twee bruikbare opties:

Migratie en Emulatie.

Bij migratie wordt het gearchiveerde bestand omgezet naar een nieuw bestandsformaat, wanneer het oude bestandsformaat ontoegankelijk raakt. Dit heeft als risico's dat er beschadigingen kunnen optreden of dat een deel van de lay-out verloren kan gaan. De transformatie naar een nieuw bestandsformaat moet ook vaker herhaald worden zodra het 'nieuwe' formaat verouderd. Migratie is echter niet geschikt voor software, aangezien de transformatie naar een nieuwer formaat vaak onmogelijk is.

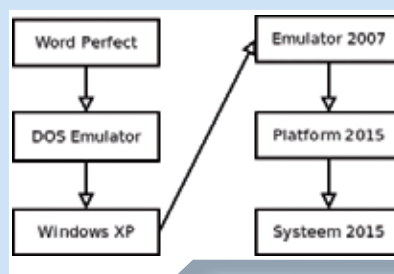
Emulatie is op het moment de enige andere archiveringsstrategie. Emulatie in deze zin houdt in dat een platform waarop de bewuste data of de specifieke software in te zien en te gebruiken is, nagebootst wordt door een ander stuk software. In tegenstelling tot migratie wordt de look-and-feel behouden in de oorspronkelijke omgeving. De twee problemen die hierbij ontstaan is het herkennen welke emulatiemogelijkheid je nodig hebt voor archivering en hoe je de emulatieomgeving zelf onderhoudbaar ontwerpt, zodat de emulator niet zelf door veroudering weg zal vallen. Op dit moment lijkt het laatste nog geen probleem te zijn, aangezien voor elk oud systeem er een emulator op internet te vinden is die op huidige systemen werkt. Het gevaar is echter dat dit over enkele jaren niet meer zo is.

Bachelorproject

Over dit probleem ging ons bachelorproject, wat we begin van de zomer van 2010 bij Data Archiving and Networked Services (DANS) [1] in Den Haag verricht hebben. DANS zorgt voor de opslag en blijvende toegankelijkheid van onderzoeksgegevens in de alfa- en gammawetenschappen. Daartoe ontwikkelt het instituut zelf duurzame archiveringsdiensten, bevordert het dat anderen dat doen, en werkt het samen met andere databeheerders. Door zijn missie is DANS van nature het Open Access principe toegedaan, maar het heeft ook oog voor het feit dat niet alle data altijd onbeperkt vrij beschikbaar kunnen zijn. Toch is het van belang dat onderzoeksgegevens die (nog) niet of beperkt beschikbaar zijn wel duurzaam worden gearchiveerd. Steeds vaker komt het voor dat niet alleen onderzoeksdata aangeboden wordt om te archiveren, maar ook software. Hierbij gaat het bijvoorbeeld om software, ontwikkeld tijdens een onderzoek, die iets specifiek met deze onderzoeksdata doet. Als opdracht vanuit DANS kregen we daarom mee: Doe onderzoek naar de haalbaarheid van emulatie als archiveringsstrategie voor software binnen DANS, en maak een prototype beheerapplicatie wat de mogelijkheden en problemen bij beheer inzichtelijk maakt.

Het probleem

Bij archivering is het voornamelijk van belang dat het archief duurzaam is. Dat wil zeggen dat zelfs met toekomstige technologische wijzigingen de opgeslagen data nog steeds te raadplegen valt. Voor de meeste data is dit geen probleem, aangezien die als enige afhankelijkheid een specifieke character-encoding heeft, maar voor software kan dit een groter probleem zijn, als deze afhankelijk is van specifieke libraries in een bepaalde Windows-versie, of van andere platformen die over lange tijd niet meer gebruikt worden.



Figuur 1: Meerdere emulatielagen

Een emulator is zelf natuurlijk ook een stuk software. Hierdoor moet er bij de emulatie-oplossing goed gekeken worden naar de emulatoren en de manier om deze ook langdurig te kunnen bewaren of gebruiken, zodat het niet of slechts beperkt nodig is om omgevingen te emuleren voor emulatoren, zoals afgebeeld in figuur 1.

Voor het bepalen of emulatie haalbaar is voor DANS, zijn een aantal vragen van belang:

- Voor welke data is het echt nodig?
- Wat zijn op dit moment de problemen?
- Welke oplossingen zijn er of worden er in de nabije toekomst afgemaakt die voor deze data werken?
- Welke juridische en financiële haken en ogen zitten er aan de mogelijke vormen van emulatie?

Voor welke data is het echt nodig?

In de introductie werd het al aangestipt: Software (en andere dynamische objecten) is vaak niet te migreren. Daardoor is Emulatie geen keus, maar een must wanneer dynamische objecten gearchiveerd moeten worden. Emulatie is hiernaast een generieke oplossing die werkt voor zowel dynamische als statische objecten, maar voor statische objecten kan migratie handiger zijn als het toegankelijk moet blijven voor een grote groep mensen.

Wat zijn op dit moment de problemen?

Voor het archiveren met emulatie hebben we een aantal problemen geïdentificeerd. Ten eerste dat van standaarden. Er zijn legio pakketten zoals VMware, Virtual-BOX e.d. beschikbaar waarmee een systeem of operating system geëmuleerd kan worden. De resulterende emulatoren zijn echter veelal niet uitwisselbaar tussen de verschillende softwarepakketten, doordat elk een eigen standaard gebruikt. Ook breder gedragen standaarden kunnen in het geding komen, waardoor het gebruiken van een 'standaard' geen garantie is voor de tijd gedurende welke de opgeslagen gegevens nog met deze standaard in te zien vallen.



Ten tweede ontbreken op dit moment de off-the-shelf oplossingen voor emulatie, wat hoge opstartkosten veroorzaakt op het moment dat emulatie gekozen wordt als archiveringsstrategie. Ook beschikbare tools zoals PRONOM[2] missen nog een duidelijke manier waarop er geautomatiseerd gebruik van gemaakt kan worden.

Ten derde is het moeilijk een verouderd systeem achteraf te gaan emuleren. De hardware waar het verouderde systeem voor geschreven was is mogelijk niet meer beschikbaar, en de specifieke drivers of ondersteunende software worden misschien ook niet meer aangeboden.

Bestaande oplossingen (in ontwikkeling)

Voor de oplossingen is gekeken naar op Europees niveau lopende projecten en samenwerkingsverbanden. Dit bracht ons in een doolhof van projectafkortingen. Ten eerste het PLANETS project (Preservation and Long-term Access through Networked Services)[3], sinds juni 2010 de Open Planets Foundation, waarin een framework in ontwikkeling is voor het aanbieden en testen van emulatietools. Hierbinnen loopt ook het KEEP-project (Keeping Emulation Environments Portable)[4] dat zich richt op het onderhoudbaar houden van emulatoren of een emulatieframework door ze makkelijk over te kunnen zetten naar andere systemen. Hierbinnen wordt gebruik gemaakt van het Universal Virtual Computer (UVC)[5] principe - een virtuele machine die zonder veel moeite op alle systemen kan draaien - om de emulatoren ook toegankelijk te houden op lange termijn. Ook wordt voor de bestandsherkenning en voor het vaststellen van de eisen voor archivering gebruik gemaakt van DROID[6] en PRONOM, twee andere onderliggende projecten.

Een oplossing voor het probleem van de off-the-shelf oplossingen wordt benaderd door de zogeheten View Paths, en een Preservation Manager in het KEEP project. Ten eerste View Paths. Dit zijn paden door een 'emulatieboom' heen, waarbij elk afzonderlijk pad één manier representeert om het stuk software of data te archiveren. Zie voor een voorbeeldboom Figuur 2. In figuur 2 is te zien dat er verschillende manieren mogelijk zijn om met de beschikbare systemen en emulatoren het stukje software te kunnen blijven gebruiken. Op het moment dat veel belangrijke data gearchiveerd worden wil je geen single points of

failure hebben, en kun je door te kijken naar de (hoeveelheid) beschikbare view paths ervoor, beredeneren of er actie moet worden ondernomen om het archief te kunnen blijven garanderen.

De Preservation Manager in het KEEP project gebruikt deze View Paths. Het onderzoekt de mogelijkheden en vereisten voor het emuleren van een bepaald stuk software of data, geeft hierover een conclusie over of iets te emuleren is, en wat er moet gebeuren voordat het geëmuleerd kan worden, mocht het niet zo zijn.

Het gebrek aan standaarden kan omzeild worden door het UVC principe toe te passen, of gebruik te maken van emulatieframeworks die dat al doen, zoals PLANETS en KEEP als onderdeel daarvan. De verwachting is dat deze emulatieframeworks binnen de archiveringswereld breed gebruikt gaan worden - omdat vele partijen er nu samen aan werken - en daardoor vanzelf een standaard worden.

Juridische en financiële issues

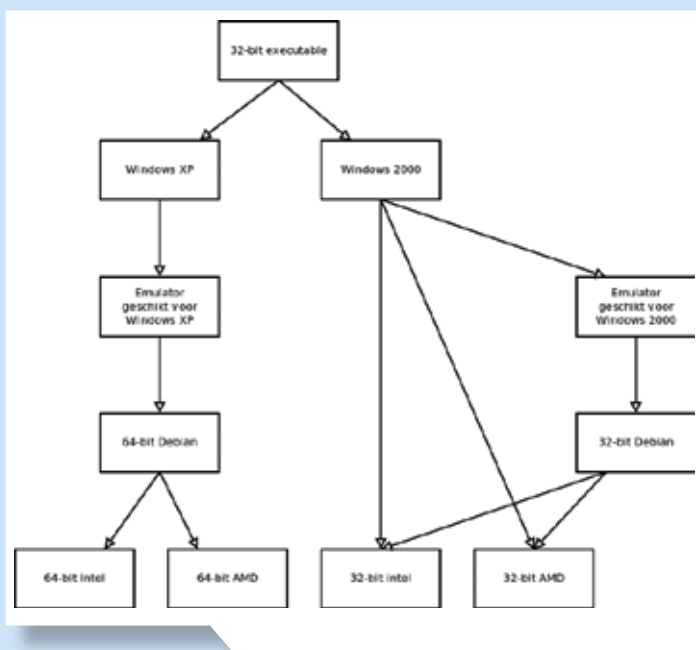
De kosten voor Emulatie zijn niet per definitie hoger dan voor andere archiveringsstrategieën. De Emulatiekosten zijn voornamelijk eenmalig, bij het maken van nieuwe emulatoren, terwijl de kosten van migratie periodiek zijn. Dit zorgt ervoor dat voor grote hoeveelheden gearchiveerde objecten het op lange termijn goedkoper is om met emulatie te archiveren, zelfs voor objecten die gemigreerd zouden kunnen worden.[7]

Tenslotte zijn de licenties een groot probleem voor het toegankelijk maken van oude data door middel van emulatie. Dit zal echter in mindere mate optreden bij het archiveren van software, omdat de eigenaar van die software toestemming zal geven voor emulatie. De platformen die hiervoor geëmuleerd worden zijn daarentegen weer wel onderhevig aan zulke licenties. Dit probleem was te uitgebreid om binnen ons project op in te kunnen gaan en er is op dit moment nog niet afdoende onderzoek naar verricht. Dit valt ook terug te zien in onze aanbevelingen.

Wat moet er nog gebeuren? Conclusies.

Er vallen de aankomende tijd een aantal mogelijkheden te identificeren die kunnen leiden tot een goede archiveringsmethode met emulatie. Ten eerste kunnen beschikbare tools zoals PRONOM breder toegankelijk gemaakt worden met een API die gebruikt kan worden door externe projecten. Er zijn, zoals eerder genoemd, veel tools beschikbaar die nuttig zijn, alleen de manier waarop ze te bereiken vallen laat te wensen over. Dit ligt ook in lijn met het standaardiseren van tools. Er worden nu al wrappers voor bestaande emulatoren gemaakt waardoor ze binnen het PLANETS framework passen. Hierdoor worden deze emulatoren op een standaardmanier benaderbaar, wat hierdoor langer mee zal gaan. Het KEEP-project biedt hier dan de definitieve en meest veelbelovende mogelijkheid om de toegankelijkheid van emulatoren op de lange termijn te waarborgen, en hiermee emulatie als archiveringsstrategie te valideren. Hierbij kan dan een remote access framework zoals GRATE gebruikt worden voor het inzien van bestanden door externen.

Binnen een hierboven beschreven systeem kunnen emulatoren blijven werken, maar hier is het aanmaken van de emulatoren dan nog een punt. Als pas na lange termijn oude systemen gepoogd worden met emulatie toegankelijk te maken, zullen sommige drivers of specifieke ondersteunende software niet meer beschikbaar zijn. Hierdoor is het van belang om emulatoren te maken terwijl de huidige systemen nog net ondersteund worden. Dit verlaagt de kosten die benodigd zijn voor het succesvol creëren van een geschikte emulator.



Figuur 2, een versimpelde emulatieboom met meerdere view paths voor een 32-bit executable.

Uitgebracht advies

Het advies wat wij aan DANS hebben uitgebracht is dan ook om vooral niet zelf eigen emulatieoplossingen te ontwikkelen, maar zich aan te sluiten bij de Open Planets Foundation en mee te gaan helpen aan het ontwikkelen van gestandaardiseerde emulatie tools die nodig zijn. Een tweede advies is om met het archiveren van software nog een tijdje te wachten tot de tools die nu in ontwikkeling zijn, af zijn. Het derde advies is om in vervolgonderzoek aandacht te besteden aan licenties bij emulatie, en aan 'Open Access', omdat het mogelijk maken van externe toegang tot de beheerde bestanden van DANS een belangrijk doel van DANS lijkt te zijn.

Over het project

Waar veel bachelorprojecten een uitvoerend karakter hebben, lag bij dit project de nadruk op onderzoek. We hadden in het begin zelfs puur een onderzoekende opdracht, zonder prototype, maar omdat een bachelorproject ook een implementatiedeel moet hebben is dat gedeelte er later aan toegevoegd. Het prototype wat we gemaakt hebben was ook het minder interessante gedeelte van het project, aangezien dat voornamelijk het vertalen van onze opgedane kennis naar een prototype inhield, wat slechts enkele keren gebruikt zou gaan worden. Dit prototype heeft aangetoond dat de meeste afwegingen uit handen van de systeembeheerders genomen kunnen worden waardoor deze zich kunnen richten op het fysieke beheren van de machines. Al met al hebben

we enorm veel plezier gehad door de ontspannen werksfeer binnen DANS, de veelzijdigheid aan collega's en de vrijheid die we kregen in het uitvoeren van de opdracht.

Het gehele onderzoeksverslag en aanbevelingsrapport valt te vinden op http://dans.knaw.nl/sites/default/files/file/Stages/Verslag_Software_Archivering_met_Emulatie.pdf

Referenties

- [1] <http://dans.knaw.nl/>
- [2] <http://www.nationalarchives.gov.uk/PRONOM/>
- [3] <http://www.planets-project.eu/>
- [4] <http://www.keep-project.eu/>
- [5] J. Rothenberg, Ensuring the longevity of digital information, Scientific American, 1995.
- [6] <http://droid.sourceforge.net/>
- [7] E. Oltmans and N. Kol, A comparison between migration and emulation in terms of costs, RLJ DigiNews, 2005.

DANS

Ken je studieadviseurs!

Jolien Kooijman, Esther Janssen & Gytha Rijnbeek

Xanvier Brouwer & Merel Stout

Hoeveel studieadviseurs zijn er op EWI actief?

Op EWI zijn we op dit moment met zijn drieën: Jolien Kooijman, Gytha Rijnbeek en Esther Janssen.

Wat is jullie functie op de faculteit?

Als studieadviseur proberen wij studenten zo goed mogelijk door hun studie heen te begeleiden. We hebben een soort intermediaire functie en werken zowel voor als achter de schermen. Zo hebben wij een voorlichtingstaak tijdens opendagen maar ook een adviserende rol in bijvoorbeeld de examencommissies en we hebben natuurlijk ook heel veel één op één gesprekken met studenten.

In principe kunnen studenten echt met alles wat hun studie beïnvloedt bij ons terecht.

Zijn jullie gekoppeld aan één opleiding?

Nee, het werk is compleet uitwisselbaar. Het kan dus ook goed zijn dat je als student de ene keer een gesprek hebt met de één en de volgende keer met een andere studieadviseur. Alleen als je een specifieke voorkeur hebt voor iemand, of indien je vervolgafspraken maakt, krijg je wel steeds dezelfde studieadviseur.

Wat is het leukst en wat is het minst leuk van de functie?

Het persoonlijk contact met de studenten vinden we veruit het leukst. Minder leuk is het als er onduidelijkheid op de faculteit heerst. Je kunt dingen beter meteen goed doen, dan dat je achteraf moet lijmen! Het tijdig ophelderen van de onduidelijkheid is wel een mooie uitdaging voor ons.

Wat is de makkelijkste manier voor een student om in contact te komen met jullie?

Er zijn meerdere manieren om met ons in contact te komen. Het beste is om een afspraak te maken via het servicepunt EWI, dan weet je namelijk zeker dat we tijd voor je hebben als je langskomt. We hebben per studieadviseur ongeveer vijf gesprekken er dag. Als we geen gesprek of vergadering hebben, staat de deur altijd open voor studenten. We willen dat studenten weten dat ze welkom zijn en streven ernaar de drempel om bij ons langs te komen zo laag mogelijk te maken. In drukke periodes hebben we tegenwoordig een open spreekuur. Het tijdstip hiervan verandert per week en bij het servicepunt kunnen ze je precies vertellen wanneer het eerstvolgende spreekuur is. Dit spreekuur is met name voor korte en voor dringende vragen zodat we zeker weten dat studenten op tijd geholpen kunnen worden.

Met wat voor vragen kunnen studenten terecht bij jullie?

Je kunt bij ons terecht voor alles wat je studie zou kunnen beïnvloeden. Denk bijvoorbeeld aan het maken van een planning voor komend jaar, maar ook problemen in de privésfeer die je studievoortgang kunnen belemmeren. Het is verstandig om ons zo veel mogelijk op de hoogte te brengen als er wat speelt, want als we op tijd weten wat er aan de hand is dan kunnen we daarop inspelen en meedenken over oplossingen/mogelijkheden. Overigens zijn al onze gesprekken met studenten vertrouwelijk.



Is er met de invoering van het Bindend Studie Advies meer behoefte gekomen aan studieadviseurs?

Of er door het Bindend Studie Advies meer behoefte is gekomen aan studieadviseurs durven we niet te zeggen. We hebben wel het idee dat de agenda's nu voller zitten met eerstejaarsstudenten. Dit komt deels omdat wij iedereen met een twijfel of negatief advies zelf uitgenodigd hebben voor een gesprek, maar het lijkt ook alsof eerstejaars sneller langs komen om te praten over bijvoorbeeld studieherkeuze en over mogelijkheden om toch nog 30EC te halen. Het invoeren van de Harde Knip zorgde er echter wel heel duidelijk voor dat er meer studenten uit zichzelf bij ons langs kwamen voor een gesprek.

“Je kunt dingen beter meteen goed doen, dan dat je achteraf moet lijmen!”

Als jullie het op de TU voor het zeggen hadden, wat zouden jullie dan willen veranderen?

We zouden dan de herkansingen niet meer in augustus willen plannen. Zoals het nu gaat is echt niet praktisch met het oog op het BSA en de Harde Knip. En we vinden dat er meer gelachen moet worden!

Wat hebben jullie zelf gestudeerd en waar hebben jullie gewerkt voordat je als studieadviseur op EWI aan de slag ging?

Jolien: ik heb menstechniek interactie gestudeerd. Daarna heb ik bij Philips Research gewerkt en nu ben ik alweer 4,5 jaar actief als studieadviseur bij EWI.

Esther: ik heb zelf geen technische achtergrond maar heb geschiedenis gestudeerd. Ik ben nu sinds september 2009 studieadviseur op EWI, daarvoor heb gewerkt bij DelftTopTech en binnen het postacademische onderwijs.

Gytha: ik loop al sinds 1993 rond op de TU Delft. Eerst als studente Industrieel Ontwerpen, daarna ben ik in 2008 teruggekomen als studieadviseur en werk hier dus ondertussen alweer 2,5 jaar. Hiervoor werkte ik als ontwerper binnen de recreatiebranche.

Missen jullie je eigen studententijd wel eens?

We missen het soms een beetje, omdat vooral de sfeer erg goed is, maar we blijven wel op gepaste afstand! Jolien heeft vorig jaar zelf nog een opleiding gedaan. Het is leuk dat je via je werk aan je studententijd herinnerd wordt, maar we hebben niet echt heimwee, we hebben er toen alles al uitgehaald! We zijn allemaal wel erg verknocht geraakt aan EWI, studenten zijn sowieso een erg leuke doelgroep. Wat ook erg leuk is, is dat we deze week [de Owee red.] samen met alle studieadviseurs een rondje langs de studentenverenigingen gaan maken.

Wat zijn jullie hobby's?

Jolien: ik hou van volleyballen, sudoku's, en knutselen. Zo maak ik bijvoorbeeld zelf knuffels. Vorig jaar heb ik de opleiding docent Nederlandse Taal gevolgd, waar ik het toen erg druk mee had, maar die heb ik nu afgerond.

Esther: mijn grootste hobby is heel lekker en uitgebreid koken, dat vind ik echt leuk om te doen.

Gytha: ik ga vaak met de decoupeerzaag aan de slag, ik maak onder andere meubels, zoals nachtkastjes. Ik vind het vooral erg leuk om creatief bezig te zijn, ik krijg voor mijn verjaardag ook altijd gereedschap. Ook ga ik graag sambadansen.

Zijn jullie al klaar voor het nieuwe collegejaar?

We zijn er nog niet helemaal klaar voor, maar dat ben je eigenlijk nooit, we hebben er in ieder geval al wel veel zin in. Je voelt nu dat er een bepaalde vibe door de faculteit gaat omdat alles weer op gang begint te komen.

Hebben jullie tot slot nog een paar laatste wijze woorden die jullie de studenten mee willen geven?

Geniet en kom op tijd naar ons toe. Je kunt heel veel dingen combineren, maar de grootste valkuil is dat studenten vaak denken alles nog wel in te kunnen halen. Je moet soms keuzes maken, anders haal je helemaal niks. 

“Je moet soms keuzes maken, anders haal je helemaal niks.”

Van links naar rechts: Gytha Rijnbeek, Jolien Kooijman & Esther Janssen





THE VIP

A FILM BY FLOW ■ TRADERS

ABOUT A VERY IMPORTANT PROGRAMMER

FLOW TRADERS IS AN INTERNATIONAL LEADER IN ELECTRONIC ARBITRAGE TRADING AND MARKET MAKING, AWARDED AS THE EUROPEAN ETF MARKET MAKER OF THE YEAR. FLOW TRADERS MONITORS INTERNATIONAL STOCK MARKETS, AND SEIZES UPON OPPORTUNITIES THROUGH INTELLIGENT ELECTRONIC ARBITRAGE PROCESSES, QUICKLY, EFFICIENTLY, AND VIRTUALLY WITHOUT RISK. FLOW TRADERS DISTINGUISHING ITSELF WITH RAZOR-SHARP TECHNOLOGY AND DEVELOP CUTTING-EDGE SOFTWARE TO STAY AHEAD OF THE COMPETITION. BY FOCUSING ON SPEED IN MARKETS WHERE EVERY SECOND COUNTS. HAVE YOU ALWAYS WANTED TO DESIGN, IMPLEMENT AND OPTIMIZE TRULY UNIQUE APPLICATIONS? AND DO YOU HAVE A KNACK FOR TRANSLATING BUSINESS REQUIREMENTS TO WORKING SOLUTIONS?

IN OTHER WORDS: ARE YOU OUR NEW VIP? VISIT WWW.VIPTHEMOVIE.COM OR VISIT ONE OF OUR IN-HOUSE DAYS!



Gadget

Dagelijkse beslomeringen

Keorb Redno

07:30

ZzzZz...*beep*beep*...oempf wat vraagt daar de aandacht?! ...*BEEP*BEEP*... Met een vlugge armbeweging richting deze bron van ellende geef je de wekker een schouderklopje dat hij je ook vandaag weer op dit onstudentelijke tijdstip wakker heeft weten te krijgen. Na een vlugge blik op je agenda op je handheld om te kijken hoe laat je waar geacht wordt aanwezig te zijn, ren je vlug even onder de douche door om jezelf op te frissen.

08:00

Aangekleed en wel beweeg je jezelf richting keuken om de belangrijkste maaltijd van de dag te bereiden en te genieten. Vandaag, heb je jezelf gisteren in de supermarkt voorgenomen, heb je een speciaal ontbijt verdiend. Na een kwartiertje bakken zit je aan de keukentafel met een lepel wat stroop, uit het potje vervaardigd door 'rinse', over je stapel pannenkoeken te smeren. Tijdens het verorberen van deze ontzettend zware bodem scan je op je handheld even snel door alle headlines van vanochtend en door de resultaten van de beurzen in andere delen van de wereld.

08:30

Ietwat tē voldaan is het tijd om naar het werk te gaan. Net voordat je de deur uit stapt herinner je je dat een huisgenoot een kist appels van z'n ouders voor z'n verjaardag heeft gekregen. Je hoort diezelfde huisgenoot nog steeds zeggen: "Mooi is dat, m'n moeder scheept me op met veertien kilo appels. Schroom jullie niet om me van deze ellende af te helpen!", dus besluit je om er maar eentje mee te pikken voor onderweg.

08:52

Nog voor je de bus in stapt, zie je een bekende van je middelbare school. Omdat je deze persoon toch al zeker zes jaar niet gesproken hebt is er natuurlijk veel te veel om bij te praten dan de korte poos in de bus jullie toestaat. Dit doet jullie overgaan tot het maken van een lunch afspraak, plots pak je jouw handheld er weer bij om een datum te prikken. Als drukke personen heb jullie pas over een aantal weken beiden een gaatje in jullie agenda's om af te spreken. Hoewel het ver van nu is, is het in ieder geval gelukt!

10:30

Eindelijk op de universiteit aangekomen voor je eerste college van de dag -saai- begint het al goed, je hebt geen flauw idee wat de docent vorige week heeft staan uitleggen. Vlug pak je de handheld er bij om de collegeslides en je notities van vorige week erop na te slaan, maar al snel dwalen je gedachten af naar het zoeken naar 'alternatieve' 'achtergrondinformatie'. Zo leer je bijvoorbeeld dat het vandaag de verjaardag is van de dame die naast je zit en begin je haar te overtuigen dat het altijd zo is geweest dat een jarige taart meeneemt voor iedereen die daar maar om vraagt. Na een aantal minuten samen gelachen te hebben zit het college er al weer op, balen, weer kostbare informatie langs je heen gegaan. Maja dat komt volgende week wel weer, want wie dan leeft wie dan zorgt.

12:30

Eindelijk lunchtijd! Hoewel je eigenlijk niet zoveel trek hebt door het nogal zware ontbijt, besluit je toch om mee te gaan naar een lokale slager om een lekker belegd broodje naar binnen te drukken. Waar zat de slager ook al weer, iedereen kijkt naar jou, maar jij zou nog geen speld in een speldenfabriek kunnen vinden dus na een vlugge blik op je handheld weet je het: 2x links, 3x rechts en anderhalf keer rechtdoor.

14:52

Na dezelfde lange weg terug naar huis, waarin je jezelf hebt vermaakt door een spelletje te spelen op je handheld, is het tijd om een start te maken met de vooravond van je afstudeerstage. Je pakt de handheld erbij en begint literatuur te zoeken voor je literatuurstudie.

18:00

Na een lange middag literatuur zoeken (lees: tower-defence spelen op je handheld), heeft je huisgenoot (die van de appels) gekookt en prijst jouw handheld omdat hij die vanochtend tijdens je douchesessie even snel geraadpleegd heeft om een recept op te zoeken.

19:30

Zo, de afwas zit erop, het is nu tijd om uit te buiken van de voortreffelijke maaltijd die je huisgenoot je heeft voorgeschoteld. Uitbuiken is natuurlijk geen uitbuiken zonder op z'n 'al-bundies' een film te hebben gekeken, daarom pak je de handheld er weer bij om je HTPC aan te zwengelen en 'Forrest Gump' aan te zetten.

22:00

ZzzZz...whuh wat?... Oef het is je weer overkomen, dit is al de derde avond op rij dat je in slaap bent gevallen voor de televisie. De film is inmiddels afgelopen en je herinnert je opeens weer dat tentamen van komende week. Na het tandenpoetsen ga je in bed liggen met je handheld en begint collegeslides door te nemen in de hoop dat je het tentamen dit keer wel haalt.

00:30

Oef taaie sessie collegeslides, niet doorheen te komen! Een korte blik op de klok verraadt dat je over 7 uur al weer wakker moet worden, dus je hangt je handheld aan het sap, stelt je wekker in en denkt nog eens na over die leuke jarige griet van vandaag.

ZzzZzzz....

Het kan gezegd worden dat bijna iedereen die er nog nooit mee in aanraking is geweest zich er wel eens negatief over heeft uitgelaten, de iPad. Deze nieuwe lijn producten van het grote fruitmerk is mikpunt geweest -en is dat stiekem nog steeds- van een overvloed aan controversie. Is het nu wel of niet een toevoeging aan je leven? Dat is mijn uitdaging voor de lezer: lees dit artikel eenmaal opnieuw en lees overal waar 'handheld' staat 'iPad'. 

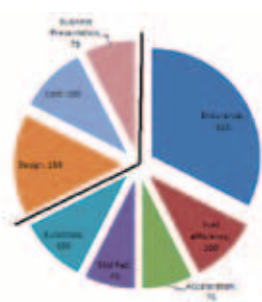
TU Delft Dreamteam

DUT Racing

Tristan Timmermans (Chief Electronics DUT10)

Wat is formula student?

Formula student is een ontwerp competitie waarvoor studenten van zo'n 400 universiteiten over de hele wereld een auto bouwen aan bepaalde specificaties die door de SAE (Amerikaanse ingenieurs vakbond) worden aangeleverd. Deze worden in Europa aangevuld met iets strengere regels. Binnen deze regels, die bijvoorbeeld voorschrijven dat de auto die je ontwerpt vier wielen moet hebben met een minimale wielbasis van 1300 mm en een maximale totale cilinderinhoud van 610cc, moet je blijven. Je bent dus vrij te ontwerpen wat je wilt en daarvoor zie je veel verschillende auto's: van 4 cilinder 600cc auto's met zo'n 100pk van 210 kilo tot kleine 1 cilinderauto's van maar 145 kilo tot aan auto's met vleugels en grote diffusers. Het mag allemaal en dat maakt de competitie zo veelzijdig: Er is geen enkele goede oplossing.



Figuur 1: puntenverdeling op de Formula student competities

De evenementen waarop de werkelijke krachtmeting plaatsvindt, bestaan uit twee soorten onderdelen: de statische onderdelen en de dynamische. De statische onderdelen zijn er om je kennis, het ontwerp van je auto en de vaardigheden van je team te testen. Het 'cost' onderdeel bestaat uit het verklaren van de kosten van je auto, want je ontwerpt hem voor een oplage van 1000 auto's per jaar en deze mag dan ook niet te exorbitant duur uitvallen. De auto van de TU kost in massaproductie zo'n 17000 euro.

Daarnaast is er ook een 'design' onderdeel. Hierbij moet je aan een team van judges uitleggen waarom jouw ontwerp zo goed is als jij denkt dat het is en waarom je bepaalde keuzes wel, of niet, gemaakt hebt. Ook is een presentatie van jouw product aan een aantal investeerders waarmee je de winstgevendheid van 1000 auto's per jaar moet bewijzen.

Bij de dynamische evenementen wordt de auto getest op verschillende punten: acceleratie, skidpad (bochtensnelheid), autocross (hanteerbaarheid door bestuurder), endurance (uithouden van de auto over langere tijd) en fuel efficiency (brandstofverbruik tijdens endurance).

Dit allemaal komt neer in een puntenverdeling zoals je in figuur 1 kan zien. De dynamische evenementen leveren het meeste op, maar om de competitie te winnen moet je op alle onderdelen goed presteren.

DUT Racing

Het team van DUT Formula Student Racing Team bestaat uit een gedeelte kernteam, soort van bestuur, en 'departementen' met engineers. De engineers zijn degene die alle onderdelen ontwerpen en dat kan iedere student zijn, van eerstejaars tot phd. Het team bestaat uit zo'n 60 studenten uit alle faculteiten, inclusief EWI, Bouwkunde en IO. In feite kan je bij iedere departement aan de slag, maar om het ontwerp van een chassis te doen heb je wel kennis nodig zoals statica en dynamica. Binnen het team staat educatie hoog in het vaandel, en er wordt geprobeerd om je zo veel mogelijk te laten leren en bezig

te zijn met de dingen die je leert te implementeren. Dit is terug te vinden in het starten van de DREAM-Team minor voor derdejaars studenten die je ook bij DUT Racing kan volgen. Per jaar doen zo'n 15 studenten dit.

Wat deed ik daar nou weer?

Maar wat doet een informaticus bij een raceteam? Ikzelf ben begonnen met het programmeren van het dashboard en het stuurtoestel in 2009. Deze bestonden uit zelfgemaakte elektronica op basis van ARMv4 processoren en de software moest dus volledig zelf geschreven worden. Daarna ben ik met de minor verder gegaan en doorgestroomd tot Chief Electronics in 2010. Als informaticus in een electrowereld is dat wel even wennen, aangezien je aan veel meer moet denken dan alleen de software die je gewend bent te schrijven. Het embedded systems practicum is er niets bij! Het hele jaar heeft voor mij grotendeels in het teken gestaan van het ontwerpen en het maken van de auto. Daarnaast zijn de mooiste momenten uiteraard het racen op het Silverstone circuit of de Hockenheimring.

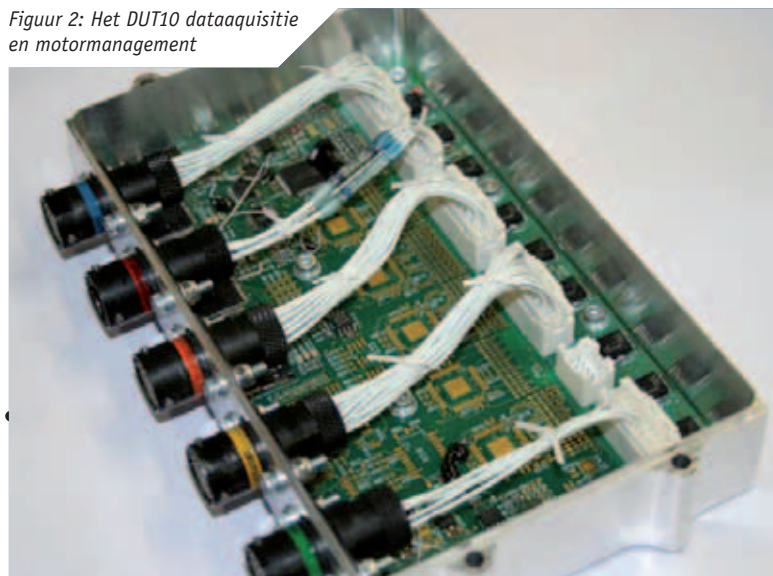
DUT10

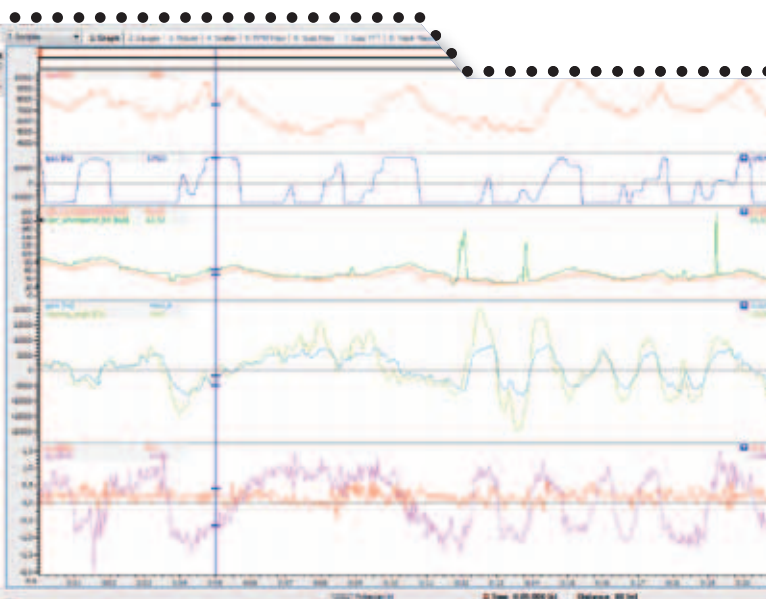
In de huidige auto, de DUT10, is alle software en elektronica zelf gemaakt. We hebben dus een motormanagement systeem dat zelf is geschreven en gemaakt, een data-aquisitiesysteem dat zo'n 40 sensoren kan waarnemen en verwerken, opslaan en weer doorsturen naar de gebruiker op 150Hz en enkele zelfontworpen sensoren. De elektronica die daarachter zit kan je zien in bijvoorbeeld plaatje [2] welke de motorcontroller en de data-aquisitie bevat. Het eigen ontwerp en de doordachtheid van de elektronica heeft ons bij de competitie in Duitsland de 'Most innovative use of electronics award' opgeleverd. Wij doen als een van de weinige teams alles zelf, omdat dit een goed leerproject is voor studenten en je de elektronica kan ontwerpen naar wat je nodig hebt en niet naar wat betaalbaar voor handen is. Ter vergelijking: Een systeem om 40 analoge sensoren te loggen kost 6500 euro.

Auto? Informatica? Wiskunde?

Wat kunnen Informatici en wiskundigen in het algemeen doen bij het team? Ik zal beginnen met een voorbeeld met wiskunde. Alles aan de auto moet doorgerekend worden, maar het liefste wil je simulatie maken van bijvoorbeeld het gedrag van de dempers (schokabsorbering en stabiliteit) van de auto, of het schakelsysteem waarbij je onder hoge snelheid zo snel mogelijk wilt schakelen. Hiervoor moeten modellen gemaakt worden en deze, veelal matlab modellen, moeten geverifieerd worden op de auto. Een wiskundige kan hier prima mee overweg en kan

Figuur 2: Het DUT10 dataaquisitie en motormanagement





De data die uit de auto komt weergegeven

delen die gemaakt moeten worden en zich ook helemaal op een onderdeel storten op het ontwerpen en maken. Een informaticus komt terecht bij data: er komt veel data binnen vanuit de sensoren en deze moet met minimale systemen, op dit moment ARMv7 instructieset in de LPC1769 chip van NXP, verwerkt worden en opgeslagen worden of gebruiken voor het berekenen van het schakelmoment van de versnelling of wanneer je de brandstof in de motor moet ontsteken. Maar ook het live verzenden van data naar de kant, iets wat je vooral gebruikt bij het testen om te zien hoe de auto zich gedraagt onder verschillende omstandigheden. Veel van deze features zijn nog niet volledig af en daar kan jij dus ook aan meewerken!

DUT11

Het huidige team gaat een iets andere kant op. Een nieuw onderdeel in de formula student competitie is het elektrisch racen. Hieraan wil het team het komende jaar deelnemen. De verbrandingsmotor gaat overboord en wordt vervangen door een, twee of vier elektromotoren. Alles is nieuw en er zijn dus genoeg kansen om daar nieuwe ideeën in te stoppen en uit te werken. Iets waar je als wiskundige of informaticus mee bezig kan zijn is bijvoorbeeld het e-differentieel (verdeling van de krachten op de twee achterwielen tijdens accelereren en bochten) of het mogelijk maken van torque-vectoring ('sturen' zonder echt te sturen). Ook komen er veel regelsystemen bij omdat je de batterijlading moet controleren over de vele batterijen en vele andere dingen.



Enkele prijzen van de DUT10

Wat is er bereikt in 2010?

Zoals ik al zei hebben we de 'most innovative use of electronics award' gewonnen in Duitsland maar dat was niet alles. In Engeland hebben we de tweede plaats op het ontwerp gewonnen met een half punt achterstand op de 150. In Duitsland hebben we verder de volgende dingen gewonnen:

- Lightweight Award rewarded by Audi
- Sportmanship Award (voor het helpen van andere teams)
- Winner Autocross (enkele ronde zo snel mogelijk afleggen)
- Winner Endurance
- Winner Fuel Efficiency
- Winner Dynamics Overall
- Winner Overall

Een vrij grote lijst. Iedereen was ook moe van naar het podium lopen achteraf.

Ook niet onbelangrijk is het leren samenwerken met andere disciplines, want je moet alle onderdelen ergens in de auto kwijt en je kunt niet zomaar een beetje kostbare ruimte reserveren zonder op iemand anders zijn tenen te gaan staan.



De DUT10 op Formula Student Germany

Jij?

Wil je ook bij het team of heb je interesse om eens te komen kijken? Kom gerust langs. Dezer dagen zit er altijd wel iemand 'op het hok' die je wat kan vertellen of laten zien. Het hok zit in de hal achter civiele techniek (waar ook practicumzalen zijn) en het staat op de TU-Plattegrond aangegeven als D:DREAM-hal. Loop de trap op en ga naar het einde van de hal. De deur staat vrij zeker open. Of stuur een mailtje naar recruitment@dutracings.nl

Voor meer informatie:

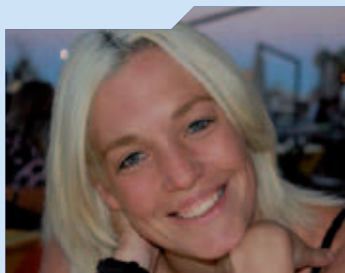
www.dutracing.nl
www.formulastudent.com
www.formulastudent.de
www.formulastudentelectric.de
<http://students.sae.org/competitions/formulaseries/>



Een kennismaking met Microsoft Surface

Freena Eijffinger

Hi, mijn naam is Freena Eijffinger en ik ben 29 jaar. In 2004 ben ik cum laude afgestudeerd aan de Hogeschool Rotterdam voor de studie Hogere Informatica. Na een aantal omzwervingen in ICT-land, ben ik in 2007 mijn eerste ICT-onderneming begonnen. Nu, 3 jaar later, staat onderneming nummer 3 opgelijnd en daar ben ik apetrots op! Naast werken heb ik ook een hobby: gamen. Ik speel een aardig potje Guitar Hero op expert level en heb daarmee o.a. de eerste prijs gewonnen op een aantal internationale wedstrijden van Microsoft. Maar naast nerd, ben ik bovenal ook de zus van een 24 jarig broertje met het syndroom van Asperger, een vorm van autisme.

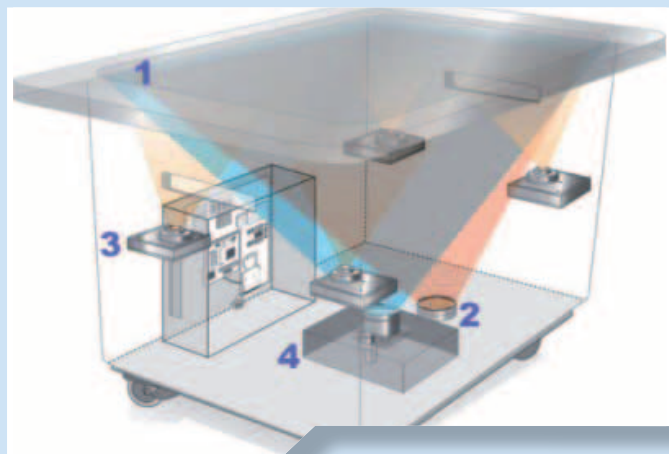


Autisme is een aangeboren stoornis in de hersenen, die onder andere invloed heeft op hoe een persoon de wereld om zich heen beleeft en hier interactie mee heeft. Het is een handicap met vele gezichten en de kenmerken van autisme zijn vaak subtiel en/of niet meteen goed zichtbaar. Wie samenleeft met iemand met autisme merkt echter al snel dat deze stoornis fundamenteel ingrijpt in het leven van, en met die persoon. Autisme heeft zowel invloed op het persoonlijk, als op het maatschappelijk functioneren van een persoon. Vaardigheden als flexibiliteit, communiceren, inlevingsvermogen en inzicht in sociale interacties zijn bij autistische mensen vaak zeer beperkt ontwikkeld. Hierdoor zijn autisten vaak hun hele leven lang in meer of mindere mate afhankelijk van anderen. In veel gevallen is het echter mogelijk om verbetering te realiseren van de vereiste vaardigheden om zelfstandig in de maatschappij te

kunnen functioneren. Deze hulpvraag en bijbehorende begeleiding zijn echter ingewikkeld en zeer persoonsafhankelijk en adequate hulp is dan ook eerder uitzondering dan regel.

Ik heb mijzelf het doel gesteld, op economisch rendabele wijze, sterke verbeteringen aan te brengen in de diagnostiek en behandel mogelijkheden van autistische stoornissen met ICT. De inzet van ICT gerelateerde middelen voor de diagnostiek en behandeling van autisme is in het verleden veelal onpraktisch gebleken wegens gebrek aan oogcontact en persoonlijke communicatie over en weer via reguliere monitoren. Een oplossing voor dit probleem trof ik aan op de Microsoft TechEd conventie in Barcelona in 2008 in de vorm van Microsoft's Surface. Nu, 2 jaar later, spreek ik zelf op beurzen over Surface, geef ik les over Surface op de Hogeschool Rotterdam en mag ik mijzelf Surface Evangelist van Microsoft noemen.

Begin 2008 komt Microsoft met haar multitouch platform: Surface. Deze computer is vormgegeven als een tafel met een scherm als tafelblad (1). Vier infrarood camera's (3) onder het scherm detecteren aanraking en geplaatste objecten en maken op die manier interactie met een gebruiker mogelijk. Een beamer (4) projecteert het beeld op het tafelbladscherm. Tevens bevat de tafel een infraroodlicht-sensor (2) die de werking van de camera's ondersteunt. De Surface tafel is bestand tegen schoppen en slaan. Hij is zelfs zo robuust, dat er op geklommen en gestaan kan worden en is bovendien spatwaterdicht.

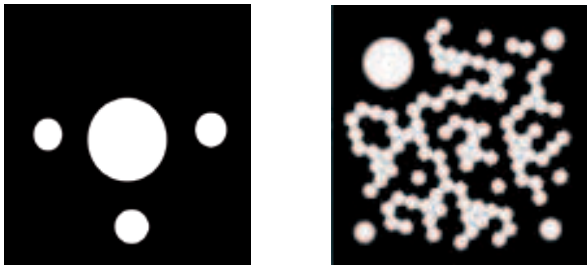


Technisch gezien is de Surface dus helemaal geen multitouch-scherm, maar een op infrarood gebaseerd systeem, dat touch ondersteunt en daarnaast ook objecten herkent. De camera's registreren wat er boven het scherm gebeurt en verwerken die data in de applicaties. Herkende objecten kunnen als volgt onderverdeeld worden:

1. Vingers; elke vinger heeft een ellipsvormige afdruk welke gedefinieerd is in de database en dus als zodanig herkend wordt. Maar de camera's registreren ook de slagschaduw van de vinger en daarmee kun je in software leuk spelen met oriëntatie.
2. Blobs; een blob is niets meer dan een herkend "iets". De camera's zien iets, maar dat iets is niet gedefinieerd, dus noemen we het een blob. Een blob heeft geen functie.
3. Shades en shapes; de camera's zien vormen en schaduwen en als je deze van tevoren definieert, kun je leuke interacties krijgen. Een eikenblad heeft bijvoorbeeld altijd, weliswaar met verschillende grootte, maar altijd dezelfde vorm. Zien jullie al leuke potenties voor een biologieles? (Denk aan verschillende boomsoorten.)



4. Byte tags: dit zijn zwart-wit stickertjes met verschillende waarden die de Surface kan uitlezen. We hebben hier 255 verschillende van (maar we hebben toch 256 bytes? – Klopt, maar 0 is niets, dus die gebruiken we niet). Deze tags gebruik je wanneer je van tevoren een maximum hebt aan te gebruiken objecten in een applicatie. Denk aan schaken of dammen bijvoorbeeld; je hebt nooit meer dan 32 schaakstukken op een bord. Stickervellen met deze tags worden geleverd bij aanschaf van een Surface unit, maar je kunt ze ook prima kopiëren met een scanner en laserprinter.



5. Identity tags: dit zijn iets complexere tags en tevens wat groter uitgevoerd. Deze tags kun je zelf maken met een softwareprogramma op de Surface unit. Hier zijn ontelbaar veel verschillende exemplaren van. Deze gebruik je dan ook, wanneer je niet van tevoren weet hoeveel tags je nodig hebt in een applicatie. Denk hierbij aan identificatiepasjes in een ziekenhuis bijvoorbeeld.

Alle applicaties dienen tenminste 52 hitpoints te ondersteunen: (4 x 2 handen) 40 vingers en een dozijn objecten. Voor een leuk filmpje over verschillende voorbeelden van Surface interacties raad ik jullie aan naar de volgende url te gaan: <http://go.microsoft.com/?linkid=9708907>

De Surface draait op Windows Vista met daarop een aparte Surface shell. Je kunt dus helaas niet zomaar een reguliere applicatie naar de Surface porteren en verwachten dat het werkt middels multitouch. De applicaties dienen echt met de Surface SDK ontwikkeld te worden. De software wordt gebouwd op het .Net platform en hiervoor maak je naast .NET gebruik van: Silverlight, WPF, XNA, XAML en Expression Blend. De Surface SDK kun je gratis downloaden via www.surface.com, als je een keer wilt stoeien met code. Via de simulator, kun je thuis op de monitor de Surface shell runnen en je applicaties testen. Ik raad eenieder echter aan, eerst de hardware te bestuderen, omdat ontwikkelen voor een NUI (Natural User Interface) echt totaal anders is dan voor een GUI (die hoef ik niet uit te leggen toch?). Waarom dan? Dat ga ik nu uitleggen.

NUI vs. GUI

Een reguliere applicatie is gericht op 1 gebruiker en wordt weergegeven op een reguliere monitor met een standaard GUI; rechtopstaand, van boven naar beneden, van links naar rechts en met mooie knoppen en pull-down menu's. Een applicatie voor hardware zoals Surface echter, is gericht op een NUI. Dit houdt in dat je gebruik moet maken van intuïtieve handelingen. Simpelweg: een kind van 3 moet het snappen, maar je oma van 80 ook...en dat zonder al teveel uitleg. Daarnaast heeft Surface een 360 graden interface, want het is een plat vlak waar meerdere mensen omheen zitten. Je hebt dus geen boven en beneden....en nog een aspect...je ontwerpt software voor meerdere gebruikers, houdt hier dus rekening mee met het bouwen van je applicatie. Pull-down menu's en knoppen zijn NIET intuïtief, hoe graag mensen die discussie ook met me aangaan....het is en blijft toch echt aangeleerd. Een kind zal een venster letterlijk van de tafel vegen, want het moet weg, maar zal nooit op het idee

komen een rood kruisje in de rechterbovenhoek aan te klikken...wil je iets groter maken dan rek je het uit met twee vingers, maar klik je niet automatisch op een vergrootglas icoontje in een menu ergens op het scherm....



Deze is dus leuk...



... en deze hebben het niet begrepen.

De afgelopen 2 jaar heb ik besteed aan het doen van onderzoek naar de mogelijkheden van Microsoft Surface tafels voor de diagnostiek en behandeling van autistische stoornissen en de conclusie kon niet anders zijn dan dat Surface een grote toegevoegde waarde kan hebben voor zowel de kwaliteit als, in positieve zin, de kosten van de diagnostiek en behandeling van autistische stoornissen. De Surface tafel werkt o.a. intuïtief, biedt een open interactie, werkt logisch en heeft geen toeters en bellen. Momenteel ben ik bezig met een internationale uitrol van mijn software en deze zal eind dit jaar binnen ongeveer 30 autisme instellingen als pilot draaien.

Voor handige technische artikelen en tutorials verwijst ik jullie graag nog naar:

- Expression Blend (<http://expression.microsoft.com/en-us/ee806443.aspx>)
- XNA (<http://msdn.microsoft.com/en-us/bb188199.aspx>)

Voor vragen ben ik bereikbaar op freena@freena.nl en twitter: [@freena](https://twitter.com/freena)

Partition

Bij het opruimen van mijn ladekast kwam ik onderstaande opgave tegen

Finals. Ik vind het een leuke opgave omdat de probleemstelling redelijk

laat u niet ontmoedigen.

A B C

Figuur 1

Onder de verdelingen die fijner zijn dan D en E bevindt zich een unieke die grover is dan alle andere. Deze verdeling noemen we het infimum van D en E. In figuur 2 is verdeling F het infimum van D en E.

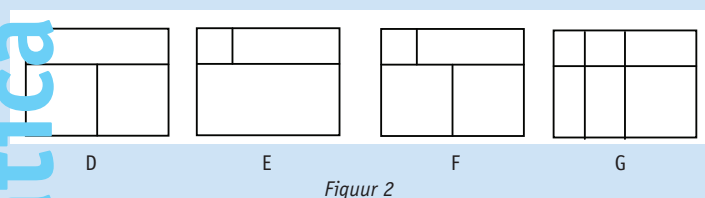
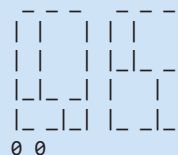
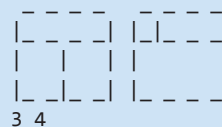


Figure 3 shows four rectangles labeled D, E, H, and K. Rectangle D is divided into two equal vertical rectangles. Rectangle E is divided into a small square on the top left and a larger rectangle on the bottom right. Rectangle H is a single rectangle. Rectangle K is divided into two equal horizontal rectangles.

Elk testgeval moet worden gevolgd door een lege regel. 

4 3



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----



Oplossing Informaticapuzzel XXIII

Quadtree codering

Dr. P.G. Kluit

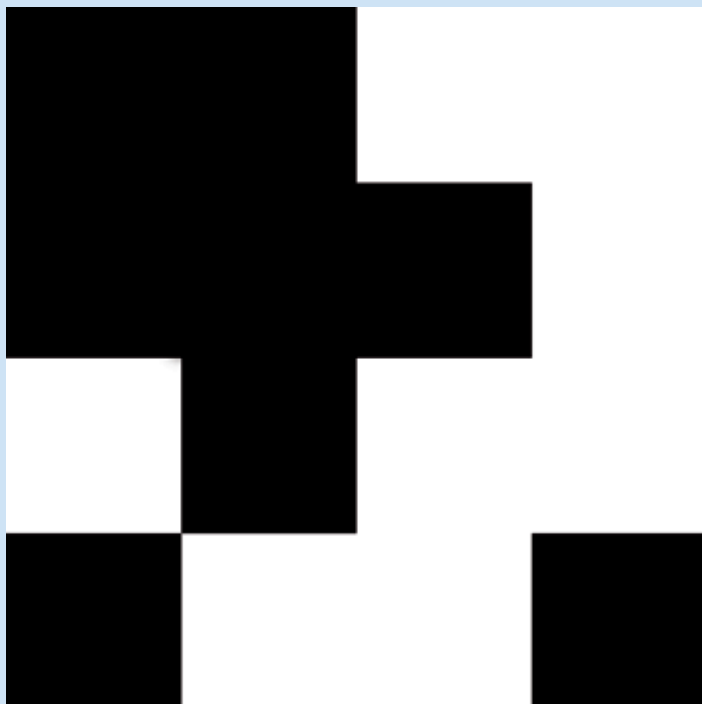
Het Quadtree probleem was relatief eenvoudig. We ontvingen oplossingen in Java van Michael de Jong, Thijmen Krebs, Xander Zonneveld, oplossingen in Haskell van Marcel Strik, Maurice Bos en Raynor Vliegendhart, en tenslotte nog een C-oplossing van Maurice Bos, waarover later meer.

Het probleem is oplosbaar door een datatype QuadTree te definiëren. Raynor Vliegendhart doet dat als volgt:

```
data QuadTree a =
  Fill a |
  Quad (QuadTree a) (QuadTree a) (QuadTree a) (QuadTree a)
  deriving (Eq, Show)
```

Vervolgens hebben we drie functies nodig: een voor het inlezen, een voor het (bottom up, recursief) reduceren van de boom, en een voor het afdrukken.

```
solve = print . simplify . read
```



Visueel voorbeeld van een quadtree codering

Het opbouwen en reduceren van de boom is wel elegant, maar ook overbodig, want je kunt gewoon op de string opereren. Alleen gaat dat niet 'bottom up'; een vereenvoudiging maakt misschien een andere vereenvoudiging eerder in de string mogelijk. Nog eens proberen dus. Dat is de oplossing van Michael de Jong:

1. If you find a square quadrant consisting of 4 black codes ('10'), replace that area with a single black code ('01'). In short:
010101010 -> 10
2. If you find a square quadrant consisting of 4 white codes ('11'), replace that area with a single white code ('11'). In short:
011111111 -> 11
3. Repeat steps 1 and 2 until we can no longer reduce the quadtree any further. Then return that reduced quadtree.

```
String reduceQuadTree(String quadTree) {
  String reducedQuadTree = quadTree;
  boolean hasBeenSimplified = true;
  do {
    reducedQuadTree =
      reducedQuadTree.replaceAll("010101010", "10");
    reducedQuadTree =
      reducedQuadTree.replaceAll("011111111", "11");

    if (reducedQuadTree.equals(quadTree))
      hasBeenSimplified = false;
    quadTree = reducedQuadTree;
  }
  while (hasBeenSimplified);
  return reducedQuadTree;
}
```

Het leuke van deze rubriek is dat je per keer de spelregels kunt aanpassen. De prijs voor de meest onleesbare (werkende) code die nog binnen een tweet past gaat dus naar Maurice Bos, voor de volgende oplossing (in C, ik zeg het er maar even bij):

```
x;_(char*a){return*a^48?a+2:9+a-x=_(_(_(a+1))))?x:(x=2[a])^4
[a]|x^6[a]|x^8[a]?a+9:2+strcpy(a,a+7);} main(int a,int*b){_(**+
b),puts(*b);} 0
```

Pattern Recognition

Hij doet het niet, of toch...?

Dr. D.M.J. Tax

Ik word toch altijd weer gelukkig wanneer ik mijn mailbox open. Niet omdat er weer een berg nieuwe vragen en problemen op me te liggen te wachten, maar omdat mijn mailprogramma alle inkomende mailtjes netjes gecodeerd heeft met vrolijke kleurtjes, gebaseerd op afzender en urgentie. Dat scheelt weer een boel zoekwerk. En het wordt toch maar eens mooi netjes automatisch voor mij gedaan.

Totdat het fout gaat natuurlijk. Zo af en toe verdwijnt er een heel belangrijk mailtje in mijn spam box. En dan ben ik later de klos, want dat zijn natuurlijk die mailtjes die een paar dagen later cruciaal blijken te zijn. Misschien is een echte butler of secretaris wel een betere oplossing voor mij, maar ja, zo'n goed salaris heb ik nou ook weer niet dus dat kan ik nooit betalen. Nee, ik moet het doen met de computer. En dat is toch wel het allermooiste van een computer, dat je die, als je 'm goed africht, al je saaie en tijdrovende klusjes kan laten doen. Geweldig! Kan ik lekker voor de TV hangen. Maar wat lekker werkt voor mijn email, werkt voor geen meter voor de fotos die ik gemaakt heb tijdens mijn vakantie (uhm, conferentie). Die foto programma's doen nooit wat ik wil. Het beste is om alle foto's op te slaan in een folder met de datum waarop de foto gemaakt is. Maar dat is behoorlijk onhandig wanneer ik die geweldige foto van de Hagia Sophia in Istanbul wil terug vinden. Wanneer was ik ook al weer in Istanbul? Ik onthou niet wanneer ik precies in Istanbul was. Maar dat uitzicht, die geweldige indruk van de Hagia Sophia, dat herinner ik me nog wel. Waarom wil die stupide computer dat dan niet?

Dit intrigeert me al een groot gedeelte van mijn leven: mensen kunnen zulke magische dingen. Jij, als persoon, kunt gewoon een onbekende kamer binnengaan, en met een enkele blik weet je waar je kan lopen, waar de tafels en stoelen staan, en waar je kunt gaan zitten. Met één enkele blik! Of je kan naar een plaatje kijken en beslissen dat het hier om een 'tafel' gaat (zie figuur 1). En mijn computer kan me alleen maar vertellen wanneer de foto is gemaakt. Lekker belangrijk...

Figuur 1: Dit zijn allebei tafels, maar waar kan je dat eigenlijk aan zien? Omdat ze allebei roze zijn?

Het blijkt dat de eenvoudige regels die in mijn mailprogramma gebruikt worden, volstrekt niet voldoen voor meer ingewikkelde dingen zoals foto's. Voor vergelijkbare prestaties moet de foto verder geanalyseerd worden. Maar ja, wat betekent dat, 'geanalyseerd'? Een van de dingen die gedaan kan worden, is om objecten uit beelden te classificeren. Dit is onderdeel van mijn onderzoeksgebied, namelijk de patroonherkenning. Geef een foto met een tafel erop en de foto wordt geclassificeerd als 'tafel'. Dit werkt echter niet met voorgedefinieerde regels, maar hiervoor moeten voorbeelden worden gebruikt. Mensen moeten voorbeelden geven van plaatjes met tafels. Uit deze beelden worden kenmerken afgeleid en met behulp van statistiek wordt een functie afgeleid die, gegeven de geëxtraheerde kenmerken, het klasse-label kan voorspellen.

Deze stap, uit de gemeten kenmerken van de voorbeelden een klasselabel voorspellen, wordt onderzocht in het vak Patroonherkennen (red. IN4085). Daarin wordt gekeken hoe het mogelijk is om uit gemeten data patronen te herkennen. Dat is uiteraard het makkelijkste wanneer je al weet waar je naar op zoek bent. Dus als je weet voor elk voorbeeld bij welke klasse het behoort. Laten we voor het gemak aannemen dat we twee klassen hebben: tafel en geen-tafel. En laten we deze twee klassen de labels $y = 1$ en $y = 0$ geven. Wanneer we nu een object hebben, meten we de kenmerken en zetten die in een zogenaamde kenmerk vector $x = (x_1, x_2, \dots, x_p)$. De patroonherkennings opdracht is nu om een classificatie functie f te vinden die zo goed mogelijk de klasselabels uit de kenmerken kan schatten: $y^* = f(x)$.

En wat betekent 'zo goed mogelijk'? Dat je elk voorbeeld goed voorspelt? Dat zou al heel mooi zijn, maar wel een beetje gevaarlijk. Want wanneer je maar een paar voorbeelden hebt, houdt niets je tegen om op basis van toevallige kenmerken te classificeren. Als er in mijn set van voorbeelden alleen maar roze tafels zitten, en alle niet-tafels hebben een andere kleur, kan ik op grond van dat kenmerk alleen al heel goed het label voorspellen. Door het feit dat ik een niet-representatieve set voorbeelden heb, ga ik op grond van foute redeneringen classificeren; ik heb een vooroordeel ontwikkeld. In de Patroonherkenning willen we daarentegen *generaliseren*: kunnen we ook een lage fout krijgen op nieuwe voorbeelden? Kunnen we niet alleen een goede voorspelling maken voor de voorbeelden die we al hebben (wat eigenlijk toch zinloos is, omdat we de labels daar toch al van hebben), maar ook voor nieuwe objecten?





Figuur 2: Links: wat voor voorwerp zou er verscholen zitten achter het zwarte vlak? Rechts: heeft iemand moeite met het vinden van de olifant?

Klasse-overlap ontstaat typisch wanneer je metingen gebruikt die niet perfect zijn. Voor het klassificeren van tafels en niet-tafels zou de hoogte van het object en het gewicht wel informatief zijn, maar er zijn natuurlijk ook objecten met dezelfde hoogte en gewicht als een tafel. Dat betekent dus dat twee verschillende objecten van twee verschillende klassen gerepresenteerd worden door twee identieke kenmerkvectoren. Bij het voorspellen van de klasse gaat er dus altijd eentje fout.

Wat nu te doen? We moeten accepteren dat er fouten optreden. We moeten beslissen hoe fout een fout is. Een standaard keuze is om de *verwachte* fout te gebruiken. De verwachte fout is het percentage van objecten dat je verwacht fout te classificeren wanneer je oneindig veel nieuwe objecten binnen krijgt. Een andere keuze is om het verwachte risico (of de verwachte kosten) te minimaliseren. Hierin wordt niet alleen geteld hoeveel objecten er fout gaan, maar ook naar welke klassen die objecten worden toegewezen. Sommige misclassificaties zijn foutter dan andere.

Stel je voor dat we onze magische functie f gemaakt hebben (hoe we dat hebben gedaan doet even niet ter zake¹). Nu staan we alleen nog maar voor de opgave om oneindig veel nieuwe test objecten te verzamelen om te zien hoe goed onze functie f eigenlijk werkt. Nu is het in praktijk al niet eenvoudig om veel voorbeelden te meten voor de constructie van f en oneindig veel test voorbeelden gaat helemaal niet lukken. We moeten het doen met een beperkte set van test voorbeelden. In praktijk wordt de set van voorbeelden gesplitst in een set die wordt gebruikt voor het fitten van f , en een set die wordt gebruikt om de fout te schatten. Zeker niet ideaal, maar beter kunnen we het tot op heden niet doen.

Dus gegeven een set van voorbeelden fitten we een functie f op één gedeelte van de voorbeelden, en vervolgens schatten we de fout op de rest van de voorbeelden. Wanneer deze fout klein genoeg lijkt, zijn we klaar, en hebben we ons classificatie probleem opgelost. Hoera! Helaas blijkt in praktijk dat dit nooit in één keer lukt. Het blijkt dat wanneer je goede kenmerken hebt, en je voldoende voorbeelden tot je beschikking hebt, de classificatie functie f geen groot probleem hoeft te zijn. We kunnen zelfs afleiden hoe de optimale functie eruit zou moeten zien. Maar daarvoor hebben we wel goede kenmerken nodig, en hoe komen we daaraan?

Het vinden en definiëren van goede kenmerken is nog grotendeels een magische kunst. Verrassend genoeg is het soms helemaal niet nodig om kenmerken van het object zelf te meten. Een voorbeeld is gegeven in het linker plaatje van figuur 2. Hier zit een voorwerp verborgen achter een zwart vlak. Voor de

meeste mensen is het vrij duidelijk uit de context dat het een kraantje moet zijn. Aan de andere kant, context kan soms erg bedrieglijk zijn, zoals te zien is in het rechter plaatje in figuur 2. Uit de context alleen zou je nooit gokken dat het voorwerp een olifant zou zijn. Context is vaak belangrijk, maar niet doorslaggevend.

Vaak kunnen wel eenvoudige kenmerken gedefinieerd worden, zoals kleur en grootte. Maar verrassend genoeg is het vrijwel nooit voldoende, en moeten er meer gecompliceerde dingen als 'vorm' en 'textuur' gemeten worden. En dit soort concepten zijn slecht gedefinieerd. Hoe beschrijf je een vorm, hoe beschrijf je een textuur? En vervolgens heb je ook nog de spatiale structuur: de tafelpoten zitten *onder* het tafelblad, en *naast* de tafel staan de stoelen. Hoe beschrijf je dat wiskundig?

Nu ben ik bij heel fundamentele, en filosofische, vragen aangekomen: wat definieert nu eigenlijk een object? Wanneer is iets een 'stoel'? Hoe het object eruit ziet? Waar het staat? Hoe het wordt gebruikt? En is de vloer onderdeel van de stoel? Kan je wel over een stoel praten wanneer er geen vloer is? Wordt iets meer een stoel wanneer het naast een tafel staat?

Op deze vragen geeft patroonherkenning (helaas) geen antwoord. Maar het geeft wel een hint wat belangrijke ingrediënten zijn, welke kenmerken van belang zijn om te kunnen voorspellen wanneer iets een stoel is of niet. En daarmee hebben we opeens iets geleerd van de computer! Dus het patroonherkennings systeem leert uit gelabelde voorbeelden, en uit het geleerde patroonherkennings systeem kunnen wij weer iets leren over de wereld.

Toch handig, die computers. 

¹ Als je geïnteresseerd bent, ben je van harte welkom bij het vak *Patroonherkennen*.



Nuna Serious Racing at the Delft Science Centre

Kees Boon

The Delft Science Centre is a new museum highlighting and demonstrating technology developed here at the TU in an entertaining way. For our Bachelor project internship, Chi-Yuen Pang, José van Amson and myself developed a Serious Game for the Science Centre Gamelab at Tinker Imagineers over the course of twelve weeks.

Serious Racing and Game Design

Our assignment stressed that the game we were to develop would need to be a Serious Game. In a Serious Game the main goal is training the user or educating them about some subject. Entertainment is only complementary to this goal, and hence not forbidden.

In addition to this constraint, we did not start out with a blank slate for our game. In a fourth year Q1 course taught by Rafael Bidarra (IN4302 Building Serious games), a prototype game had already been developed using the Microsoft XNA framework, and we were to expand on it. Some of the members of the prototype team were brought in for support and assistance.

Starting out, the game was a two player racing game. One player took on the role of 'driver', and the other player took on the role of 'designer'. While the driver raced the Zandvoort racetrack, the designer could make adjustments to the car on the fly that would immediately alter the car's driving (and aesthetic) characteristics. For an added TU Delft connection, the car in question was the Nuna solar car. This way the players had to coordinate with each other to find the best design choices for different parts of the track. Completing a lap in this way emphasized the challenge of designing a car for differing situations by weighing costs and benefits, thus providing the Serious Gaming aspect.

This led to another constraint that we had to work with: Traditionally, racing games award score based on a player's lap time. This scoring method was not to be used as it would only reward players for designing a fast car while other design choices could easily be neglected. Another way of scoring players' efforts would have to be found.

101010110010101011001010101100101010110010101011
123581321112358132111235813211123581321
101
123426789012342678901234267890123426789
1010101001101010100110101010011010101001101010100

Orientation and Planning

With the above aspects in mind, we set out to implement our vision of the game. Before we could do so though, we needed an actual vision for our version of the game, and the way we were going to create it.

Creating a game that is educational, but not boring can be helped immensely by using the Agile development method. Using this method, we required periodic user feedback so we could adjust the 'heading' of our game based on users' experiences and comments. Where a programmer's creator-bias might prevent him from noticing poor gameplay, user tests can point out such shortcomings and consequently improve the final product.

Allowing for course corrections during the twelve weeks of development, we came up with several gameplay ideas to enhance the serious aspect of the game. In the prototype version, the only car design options were color, engine and types of wheels. The choice of engine could of course change the Nuna 5 into a petrol powered car, but within a game this would not be too much of a stretch. Since these two options already allowed for quite some variation in a game that would only be played for a few minutes at most, we decided to add a single other design option; the car bodywork material.

The game can be completed using only one design option throughout, but we wanted to strongly encourage (if not force) the players to change the car design. At first we considered limiting the availability of design options based on some metric such as time or material cost, but this idea was dismissed, quite rightly, by the project support members as it would limit design choices adversely, whereas encouraging different choices was the game's intention. Enhancing the pressure to make sensible design decisions by awarding correct ones was the way to go. Hence we came up with driving challenges, that would be considerably easier in some car configurations than others. This would increase player communication, the necessity of making right design choices and make the game more fun to play. Instant rewards in the form of car upgrades would be used as an incentive for completing these challenges. We elected to include narrow 'bonus strip' sections of the track that the driver needed to follow, a time trial that would require a length of track to be completed before a timer ran out and a slalom section requiring driving agility as on-track challenges.

Some steps had already been made in the direction of decoupling scoring from time by the prototype team. An 'eco-score' measuring the car's environmental impact was already in place. We decided to overhaul this scoring system because it focused only on the environmental aspect of car design, and not on any of the myriad of other choices involved in designing a car. Fortunately, the earlier developers had taken note of this, and had already implemented some support for this which was not yet used to its full potential. Our plan was



Information



to take the environmental impact, ease of maintenance, ease of production and fuel consumption design aspects and compute a score from these four elements. Players also needed to be able to see these values change during and after the race, so they could observe the effects of their design changes.

Several recommendations for future enhancements of the gameplay were made by the prototype team. One such recommendation was clarifying on screen information, which was unclear in some cases. We chose to heed the recommendation and improved the on-screen icons and player feedback in general. Amongst other things, we elected to implement audio into the game for this purpose. Audio also improves the overall quality of a game, so that would be an additional bonus.

Coding, User Test and Architecture

Our plans laid out, we set to work. The Science Centre had originally planned to house us in its Gamelab for the duration of the project. The Gamelab is the exposition room where games developed by students are played by the visitors. In this setup, early visitors of the museum could interact with the programmers and see and play the game that was being developed. Since the Gamelab room renovation was not finished when we started the project, the logistics of this plan proved unfeasible. We ended up being housed in a meeting room together with the Robot-Pacman team for the whole project (who have their own article in this issue).

It would be near impossible to summarize the entire programming run in a few paragraphs, therefore only some parts of the development process are highlighted below.

A great deal of software architecture choices had already been made for us since we were starting out from a functional prototype. The code we started with proved to be easily extendable with the exception of a few select areas such as the network or the external physics library.

The game runs on a PC and since it is a two player game, two applications were used which communicated in a client-server way. In this set-up, the racing part fulfilled the server role, while the designer part was the client. Initially, only the client (designer) could send information on car design changes to the server (race game). This was insufficient for our plans because, for instance, information on the status of driving challenges and positional information for a map of the track with the car's position needed to be sent from the server to the client. During development we overhauled the networking code several times until it was general enough for our purposes.



Early on we set out implementing the scoring code. This proved difficult however, since ongoing changes to the game dynamics would radically change the resulting score. Particularly because we wanted to have an arcade-like score shown onscreen that would indicate score obtained thus far. The four score counting elements would be weighed with the car speed and accumulate over time. A rough version of this scoring method was used for the first user test.

Regardless of our skills as programmers and game designers, we were ill-prepared for guiding some two dozen primary school children through our test. As a consequence the tests went quite chaotically. Fortunately, the children clearly enjoyed the game and provided us with both implicit and explicit feedback. For instance, the real-time scoring method described above was almost completely ignored, so we opted to change the score calculation to be done after the lap was finished. This allowed us to give a better overview of the score and also calculate it using the information obtained throughout the entire race instead of only local data. Interviews we conducted gave us information on what HUD icons and game concepts were unclear.

Two of our three man team had a small amount of experience creating 3D and 2D art. One of the common points throughout the user feedback was that the game was somewhat empty. Using our limited skills in the graphics area, we were able to create clearer icons and add a considerable amount of 2D and 3D art to the game. Especially the designer screen received a lot of attention, because it had to be controlled using a touch-screen and therefore needed to be intuitive and controllable with relatively coarse precision. Fortunately for us, the game was not to be a graphics demonstration: it would be silly to compete with industry standards on a project like this, and the game's primary value would have to come from the Serious Gaming angle in any case.

Deployment

As noted above, the designing player would enact his or her decisions using a touch screen. This was not the only part of the end-user experience we had to keep in mind; the driver would control the car with a Microsoft Xbox360 wireless steering wheel, and both parts of the game needing to run on separate monitors in full screen for hours on end, day after day and keep malicious users from accessing the underlying system. The Gamelab houses a few dozen games, so the game needed to automatically start and keep running without Science Centre staff intervention.

Amongst other exceedingly bizarre problems that these simple requirements created, one was running two applications in full screen. As far as I knew, no commercial game but 'Supreme Commander' even had this functionality, so while the proof of concept was there, dual-fullscreen was not a common feature. Therefore it was likely not easily done. This is mostly due to Windows 7 and ATI graphics cards not liking that functionality at all. Serendipity proved to be a life saver in this case. While solving a strange video-card related crash that occurred when both parts of the game were started within too small a time of each other, we found that XNA offers a feature of faking full-screen mode. With this hacky solution working fantastically, the game was finally ready to be played by Science Centre guests.

In closing

For more information about the Science Centre, you can visit their website at sciencecentre.tudelft.nl. A (discontinued some time during development) weblog of our activities in creating this game can be found at <http://race-game2010.weblog.tudelft.nl/>



BSc Project Technische Informatica

Project Pacman

Daco Harkes

Ruim tien jaar geleden speelde ik spelletjes op een 386. Ducktales, Dune 2 en Pinball vond ik geweldig, echter bij Pacman werd ik in het 2e level altijd door die haaien opgegeten. Drie jaar geleden begon ik met mijn studie Technische Informatica in Delft. Gedreven door een passie voor computers en games zat ik daar prima op mijn plaats. Nu is mijn bachelor alweer achter de rug. Aan het einde van de bachelor kwam het bachelor project en tussen de grote lijst met projecten zat een bijzonder project; Voor het Science Centre van de TU een Pacman game klussen met radiografisch bestuurbare robots. Wat is nu mooier dan zowel een game maken als spelen met radiografisch bestuurbare robots tegelijk?

De opdracht was om het klassieke pacman spel na te maken met robots. De werking van de kunstmatige intelligentie van de computerspelers moest daarbij ook weergegeven worden. We hebben onze eigen draai aan de opdracht gegeven door Pacman opdrachten te laten doen in plaats van bolletjes eten en door de muurtjes weg te laten. Ook hebben we ervoor gekozen om naast de interne werking van de kunstmatige intelligentie, de interne werking van de beeldverwerking weer te geven.

De beeldverwerking brengt ons gelijk bij het eindproduct. Om een volledig werkend systeem met computerspelers te creëren hebben wij gekozen voor een centrale computer. Deze computer analyseert met behulp van een webcam en beeldverwerking de spelsituatie en stuurt daarna radiografisch de robots aan. Het systeem is behoorlijk veelzijdig waardoor een diversiteit van vakken uit het curriculum aan bod kwam; radiografisch bestuurbare robots (embedded systems), robots herkennen van een webcam beeld (beeldverwerking), de interface op de schermen met informatie en de Xbox controller om pacman

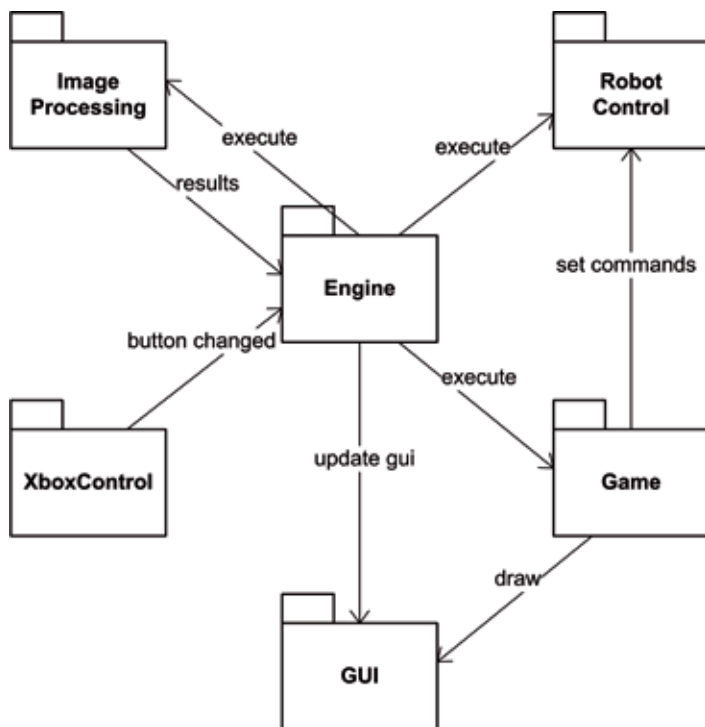
te besturen (o.a. mens-computer interactie) en de spellogica zelf (computer-games). Ook was er bijvoorbeeld pathfinding nodig om de robots correct te besturen (algoritme) en moest er behoorlijk wat denkwerk verricht worden om een goede software architectuur te creëren (o.a. software engineering).

We hebben scrum gebruikt als ontwikkel methode. Deze methode houdt in dat je elke 30 dagen naar een x aantal milestones werkt, een sprint, en elke dag een korte evaluatie hebt. Wij hebben de sprints verkort naar twee weken en de korte evaluaties vaak verzaakt. Deze ontwikkelmethode heeft ervoor gezorgd dat we elke twee weken een werkend product hadden staan en dat dan gingen verbeteren. Uiteraard veranderde gedurende het project de eisen en inzichten waardoor we de laatste twee sprints eigenlijk iets heel anders hebben gedaan dan gepland aan het begin. Dit was erg fijn voor het uiteindelijke eindresultaat.

Zoals al eerder aangestipt hebben we de muurtjes laten verdwijnen. Na dat een aantal tieners van de doelgroep langs was geweest was het duidelijk dat de muurtjes weinig speelplezier toe zouden voegen. Daarnaast werd de beeldverwerking een stuk eenvoudiger zonder muurtjes.

Om de techniek in te duiken wil ik wat van de architectuur beschrijven en inzoomen op verschillende subsystemen. De architectuur was in hoge mate modulair, elke functionaliteit zat in zijn eigen subsysteem waarvan de interface strak gedefinieerd was. Zoals te zien in het diagram staat de engine in het midden en krijgt de rest daar commando's van.





De Image Processing subsysteem gebruikt een groot aantal opeenvolgende beeldverwerkingsoperaties om de feed van een webcam te verwerken en te analyseren. Dit zijn over het algemeen basale operaties zoals de afgeleide, om de randen te detecteren, binaire optel- en verschiloperaties, opvullen, uitknippen en maskeren. Deze simpele stappen hebben we met trial en error achter elkaar gebruikt om zo stabiel mogelijk de tafel en de robots te detecteren. Op de afbeelding zie je het scherm met de debug-informatie van de beeldverwerking. Op de kleinere plaatjes, de uitgeknipte robots, zie je eerst het resultaat van de identificatie. Daarna zie je stap voor stap wat er gebeurt om de robots te identificeren. Eerst wordt er een plaatje met de randen gemaakt door middel van de afgeleide van het plaatje. Daarna wordt het plaatje gedrempeld om een binair plaatje te krijgen. Vervolgens worden de binaire vormen die achter blijven op de afbeelding geanalyseerd om vervolgens op basis van onder andere lengte/breedte-verhouding alleen de rechthoek en de cirkel eruit te filteren. Daarop worden de pixels van de rechthoek en de cirkel in het gekleurde plaatje geanalyseerd om de kleur te achterhalen en daarmee de robot te identificeren. De laatste stap is het bepalen van de positie en richting van de robots, dat wordt gedaan op basis van de zwaartepunten van de cirkels en rechthoeken.

Nadat de robots geïdentificeerd en gelokaliseerd zijn moeten ze ook aangestuurd worden. Hiervoor is het Robot Control subsysteem verantwoordelijk. Het besturen van een robot en het bijsturen op basis van de feedback via de beeldverwerking is een typisch regelsysteem. Karakteristiek ook voor dit regelsysteem is de vertraging van de commando's naar de robots en de feedback over de positie van de robots door de beeldverwerking. Daardoor was er aan het begin een hoop overcompensatie voor het bijsturen van in een rechte lijn rijden. Nadat het scheduling algoritme voor het sturen van de commando's gereviseerd was, van een queue naar een aanpassing op round robin, konden de robots een stuk nauwkeuriger aangestuurd worden. Ook bleek het achterwege laten van acknowledgements zo veel zendtijd op te leveren dat de robots duidelijk sneller reageerden. Beide keren bleek een eenvoudigere oplossing effectiever dan een geavanceerde. Centraal werden de robots aangestuurd met commando's via een command pattern. Hierdoor was het mogelijk om commando's achter elkaar

uit te voeren en zelfs een lijst commando's op repeat te zetten. Met behulp van een lijst commando's als 'rijd naar punt (x,y)' en de repeat kon binnen no-time een patrouillerende robot op tafel getoverd worden.

Naast technische details zijn er ook een hoop praktische zaken aan de orde gekomen. Op foto zie je de ontwikkellocatie tijdens een van de laatste dagen. Op de robots zitten daar nette kunststof plaatjes. Deze waren er pas in de laatste week. Dit kwam voornamelijk doordat de verantwoordelijkheden niet duidelijk waren. Gedurende het project werd duidelijk dat we het overgrote deel van de verantwoordelijkheid zelf moesten nemen, anders kon een aantal zaken wat langer duren dan gepland. Omdat we afhankelijk waren van de opdrachtgever en ontwerper om de uiteindelijke opstelling te realiseren was de opstelling nog niet klaar op de laatste dag van ons project. Met beeldverwerking die toch wel situatiespecifiek is was het ook geen succes toen de uiteindelijke opstelling in gebruik werd genomen. Gelukkig waren de problemen deels te verhelpen met een aantal simpele tweaks.

Over de invulling van het spel zelf is een hoop gebrainstormd en gediscussieerd. Dit resulteerde in een stijgende leercurve en vervelend effectieve kunstmatige intelligentie in de laatste levels. Omdat het doel van het Science Centre is bezoekers, in de eerste plaats tieners, een kijk in de techniek te geven moesten we voor de kunstmatige intelligentie gebruik maken van technieken die makkelijk te visualiseren zijn. Uiteindelijk hebben we voor een tweetal simpele state diagrammen gekozen waarbij de robots naar bepaalde posities rijden op basis van afstand tot pacman en willekeur. Niet elegant maar zeer effectief. Als een van de robots net van koers wijzigt op het moment dat je het niet verwacht, dan is er bijna geen ontkomen meer aan.

Ook is er natuurlijk de User Interface. Op de schermen kun je een soort HUD zien. Het rechterscherm toont het camerabeeld met de tafel, pacman, de spookjes, de opdracht die gedaan moet worden, de score en de highscore. Eigenlijk gewoon een standaard scherm voor een spel. Het linker scherm geeft een kijkje aan de binnenkant van het programma. Bovenaan de kunstmatige intelligentie en de pathfinding en onderaan de beeldverwerking. Bij het beeldverwerkingsgedeelte word getoond hoe het webcam beeld steeds verder verwerkt word tot een resultaat. Dit scherm deed ook prima dienst als debugscherm voor als bijvoorbeeld de beeldverwerking door andere lichtinval niet stabiel meer was.

Na twee maanden tosti-lucht, sponsored by onze racegame collega's, verhitte discussies, met dank aan mijn projectgenoten Bram, Daan en Rick, en duizenden regels code kunnen nu bezoekers van het Science Centre Pacman spelen met radiografisch bestuurbare robots. Zo heb ik mijn bachelor met een veelzijdig bachelor project afgesloten. Pacman, deze keer zonder haaien! 

Reconstructie van de aardkorst met tomografie

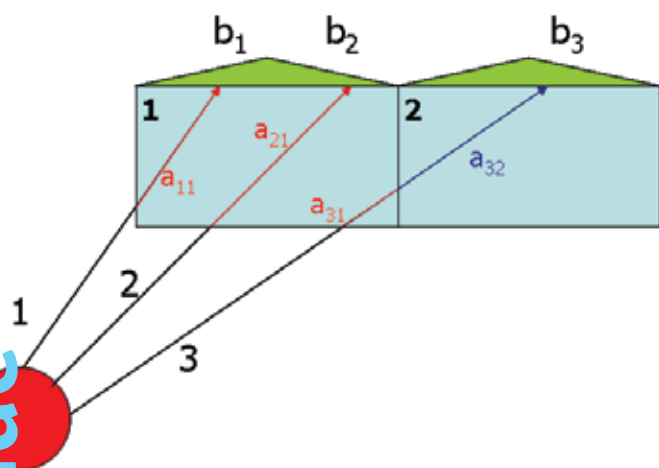
Thea Vuik

Bodemonderzoek op kleine schaal (tientallen meters diep) is eenvoudig uit te voeren: bodemonsters vertellen precies welke materialen in de aardkorst voorkomen. In diepere gebieden (tot enkele kilometers diep) kan seismiek gebruikt worden om eigenschappen van de aardkorst te bepalen. Op grotere schaal (honderden kilometers diep) zijn deze methoden niet toepasbaar. In dit geval worden aardbevingen gebruikt om informatie in te winnen. Een aardbeving zorgt dus niet alleen voor verwoesting, maar ook voor onderzoeksresultaten!

In mijn bachelorproject heb ik onderzocht op welke manier een aardbeving gebruikt wordt om eigenschappen van de aardkorst te bepalen. In dit artikel leg ik uit op welke manier ik te werk ben gegaan. Allereerst wordt op vereenvoudigde wijze uitgelegd hoe we tot een wiskundige voorstelling van dit probleem komen. Vervolgens wordt een modelprobleem gebruikt om oplossingsmethoden te bespreken. Tenslotte worden deze methoden met elkaar vergeleken.

Vertaling probleem naar wiskunde

Ieder materiaal heeft een eigen snelheid waarmee drukgolven zich verplaatsen. Als deze voortplantingssnelheid bekend is, kan het bijbehorende materiaal bepaald worden. In figuur 1 is een vereenvoudigde weergave van een probleem te zien. De blauwe rechthoek is het deel van het aardoppervlak dat onderzocht wordt. Hierbij wordt een rooster gebruikt, bestaand uit twee cellen. Op grote diepte vindt een beving plaats: hieruit gaan drie drukgolven (stralen) door het gebied. Op het aardoppervlak staan seismografen: zij meten wanneer een straal aankomt.



Figuur 1: Weergave van tomografieprobleem

De richting van de stralen wordt in dit geval als bekend verondersteld (te benaderen met stralentheorie). Als gevolg hiervan zijn de volgende waarden bekend:

- a_{ij} : de lengte van straal i door cel j , $i = 1, 2, 3$; $j = 1, 2$;
- b_i : de tijd die straal i in het gebied doorbrengt, $i = 1, 2, 3$.

Gezocht wordt naar de waarde van v_j : de voortplantingssnelheid door cel j , $j = 1, 2$. Merk op dat het voldoende is om de traagheid $x_j = \frac{1}{v_j}$ te bepalen, $j = 1, 2$.

Met behulp van de relatie

$$\text{totale reistijd} = \sum_{\text{cellen}} \frac{\text{lengte door cel}}{\text{snelheid in cel}}$$

vinden we nu dat moet gelden:

$$b_i = \sum_{j=1}^2 a_{ij} x_j, \quad i = 1, 2, 3.$$

Dit leidt tot de volgende drie vergelijkingen:

$$\begin{aligned} a_{11}x_1 &= b_1; \\ a_{21}x_1 &= b_2; \\ a_{31}x_1 + a_{32}x_2 &= b_3. \end{aligned}$$

Hierbij is de vector $\mathbf{x}$ nog onbekend. We zien dat de zoektocht naar de aard van materialen in de aardkorst leidt tot het oplossen van het stelsel $A\mathbf{x} = \mathbf{b}$!

Modelprobleem

Vervolgens wordt uitgegaan van een modelprobleem: een deel van de aardkorst (600 km breed, 300 km diep) waarvan de aanwezige materialen, de vector $\mathbf{x}_{\text{model}}$, bekend zijn. Het gebied wordt verdeeld in 200 gridcellen: 20 in de breedte, 10 in de diepte. Doel van het onderzoek is nu methoden te vinden die $\mathbf{x}_{\text{model}}$ zo goed mogelijk benaderen.

Er worden 400 metingen verricht, zodat $A \in \mathbb{R}^{400 \times 200}$. Omdat er altijd meetfouten optreden, is $\mathbf{b}$ niet exact. Als gevolg hiervan zit $\mathbf{b}$ niet in de kolomruimte van A , zodat het stelsel in principe niet oplosbaar is. De best mogelijke oplossing wordt nu gevonden door te eisen dat $\|A\mathbf{x} - \mathbf{b}\|$ minimaal is. De kolommen van A zijn over het algemeen afhankelijk, waardoor ook de kleinste kwadratenoplossing niet uniek bepaald kan worden.

Omdat A afhankelijke kolommen heeft bevat de nulruimte van A vectoren $\mathbf{x}_n$ ongelijk aan nul, zodanig dat $A\mathbf{x}_n = \mathbf{0}$. Hieruit volgt dat een gevonden oplossing niet uniek is: als $A\mathbf{x}_p = \mathbf{b}$, dan voldoet ook $\mathbf{x} = \mathbf{x}_p + \mathbf{x}_n$. Om een eenduidige oplossing te krijgen, wordt geëist dat $\mathbf{x}$ loodrecht op de nulruimte van A staat. Dit is equivalent met de eis dat $\|\mathbf{x}\|$ minimaal moet zijn. In de volgende subparagrafen worden drie onderzochte oplossingsstrategieën besproken.

Singuliere waardenontbinding

Singuliere waardenontbinding is een methode die werkt volgens hetzelfde principe als eigenwaardenontbinding bij vierkante matrices. Het is mogelijk om A te schrijven als $A = U\Sigma V^T$, waarbij $U \in \mathbb{R}^{400 \times 400}$ en $V \in \mathbb{R}^{200 \times 200}$ voor transformatie zorgen. Deze matrices zijn orthogonaal, zodat $U^{-1} = U^T$ en $V^{-1} = V^T$. $\Sigma \in \mathbb{R}^{400 \times 200}$ is een diagonaalmatrix, met $\sigma_j = \sqrt{\text{eigenwaarde } j \text{ van } A^T A}$. Hierbij kan σ_j ook gelijk aan nul zijn.

Een benadering voor $\mathbf{x}$ wordt gevonden met de pseudoinverse van A : $\mathbf{x}^+ = A^+ \mathbf{b} = V\Sigma^+ U^T \mathbf{b}$. Voor Σ^+ geldt dat $\sigma_j^+ = \frac{1}{\sigma_j}$, mits $\sigma_j \neq 0$. Als $\sigma_j = 0$, dan is $\sigma_j^+ = 0$.

Tikhonov regularisatie

Een tweede manier om het inconsistente stelsel $A\mathbf{x} = \mathbf{b}$ op te lossen, is door Tikhonov regularisatie toe te passen. Hierdoor wordt het stelsel vergroot: gezocht wordt naar $\mathbf{x}$ zodat

$$\begin{pmatrix} A \\ \lambda I \end{pmatrix} \mathbf{x} = \begin{pmatrix} \mathbf{b} \\ \mathbf{0} \end{pmatrix}$$

voor zekere $\lambda > 0$. Het is nu wel mogelijk de kleinste kwadratenoplossing te bepalen: de normaalvergelijkingen worden gegeven door

$$(A^T \quad \lambda I) \begin{pmatrix} A \\ \lambda I \end{pmatrix} \mathbf{x} = (A^T \quad \lambda I) \begin{pmatrix} \mathbf{b} \\ \mathbf{0} \end{pmatrix}.$$

Dit leidt tot $(A^T A + \lambda^2 I) \mathbf{x} = A^T \mathbf{b}$. Omdat $\lambda > 0$ is, bestaat de inverse van $A^T A + \lambda^2 I$ altijd.



ART

De derde en laatste onderzochte methode is de methode van ART (Algebraic Reconstruction Techniques). Dit is een iteratieve methode, die per vergelijking corrigeert, zodat de oplossing steeds aan één vergelijking van het stelsel voldoet. Hiervoor wordt de volgende terminologie ingevoerd:

$$A = \begin{pmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_{400}^T \end{pmatrix}; \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{200} \end{pmatrix}; \quad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{400} \end{pmatrix}.$$

Stel dat een schatting $\mathbf{x}^k$ bekend is. Deze oplossing is over het algemeen niet exact, dus

$$A\mathbf{x}^k = \mathbf{b} - (\mathbf{b} - A\mathbf{x}^k).$$

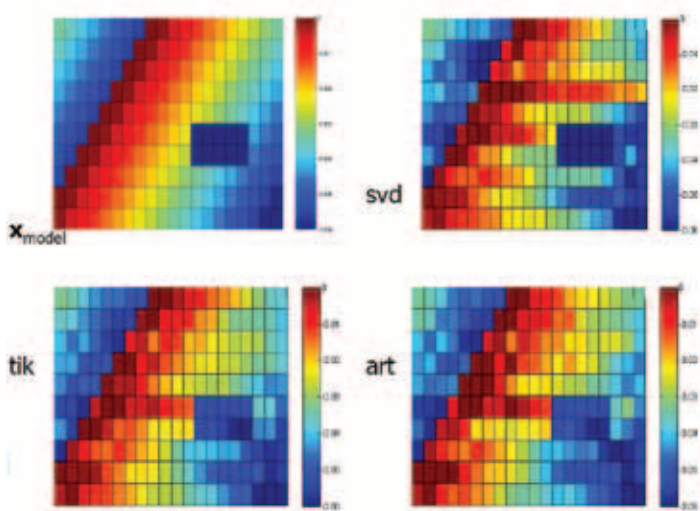
De methode van ART bepaalt nu een correctie $\Delta\mathbf{x}^k$, zodanig dat bij de volgende iteratie zo goed mogelijk aan de i -de vergelijking voldaan wordt:

$$(A\mathbf{x}^{k+1})_i = (A(\mathbf{x}^k + \Delta\mathbf{x}^k))_i = b_i.$$

Bij toepassing van ART op het probleem wordt bovenstaand principe gebruikt, waarbij iedere iteratie (het één keer doorlopen van alle 400 vergelijkingen) meer informatie aan $\mathbf{x}$ toevoegt. ART geeft al na tien iteraties een grof beeld van de gezochte $\mathbf{x}$, en voor een verfijnd beeld zijn zo'n dertig iteraties noodzakelijk.

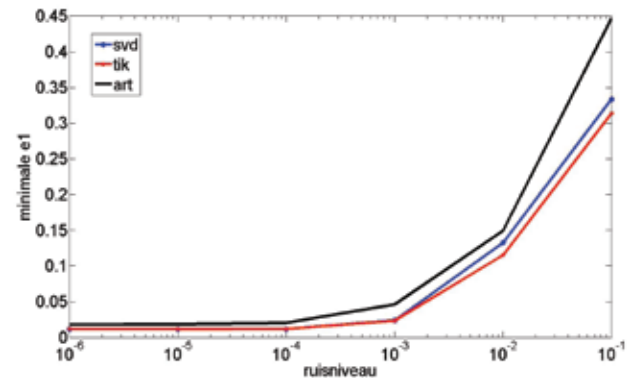
Vergelijking methoden

Bovenstaande oplosmethoden kunnen op verschillende manieren vergeleken worden. Allereerst is vergelijking op het oog mogelijk. Hiertoe wordt de gevonden $\mathbf{x}$ gevisualiseerd. In figuur 2 zijn de best mogelijke oplossingen te zien die met behulp van de drie methoden gecreëerd kunnen worden. Linksbovenin is de echte oplossing te zien. De bovenste gridcellen vertegenwoordigen het aardoppervlak en overeenkomstige kleuren zijn dezelfde materialen. We zien dat alle drie methoden een redelijk beeld geven van de daadwerkelijke oplossing.



Figuur 2: Overzicht met exacte oplossing (linksbovenin) en best mogelijke oplossing bij iedere methode

Daarnaast is het mogelijk om de methoden te vergelijken aan de hand van de gemaakte fout. Hiertoe wordt de relatieve globale fout e gedefinieerd als $e = \frac{\|\mathbf{x} - \mathbf{x}_{\text{model}}\|}{\|\mathbf{x}_{\text{model}}\|}$. Bij verschillende waarden voor het ruisniveau (grootte van afwijking ten opzichte van onverstoord $\mathbf{b}$) kan nu de minimale waarde van e bepaald worden. In figuur 3 is deze waarde voor de verschillende methoden te zien. Hieruit blijkt dat Tikhonov het best presteert: de fout is altijd het kleinst.



Figuur 3: Minimale waarde van e bij gebruik van verschillende methoden

Een derde vergelijking kan gemaakt worden aan de hand van de vereiste rekentijd. In tabel 1 is het aantal te verrichten operaties te zien bij toepassing van de methoden op het modelprobleem.

Methode	Aantal operaties
Singuliere waardenontbinding	128 miljoen
Tikhonov regularisatie	19 miljoen per λ , totaal ± 190 miljoen
ART	0.32 miljoen per iteratie, totaal ± 10 miljoen

Tabel 1: Aantal benodigde operaties voor modelprobleem

Bij Tikhonov regularisatie worden ongeveer tien waarden voor λ getest en voor ART zijn zo'n dertig operaties noodzakelijk: ART blijkt de meest voordelige methode te zijn. In de kortst mogelijke tijd wordt een goed idee van de oplossing verkregen.

Omdat de beste methode afhankelijk is van verschillende belangen (kwaliteit, tijd) kan de keuze voor een methode bij ieder probleem anders zijn.

Referenties

- [1] Nolet, Guust, 1987. *Seismic wave propagation and seismic tomography*. Seismic tomography with applications in global seismology and exploration geophysics (editor: Nolet, Guust). D. Reidel Publishing Company, Dordrecht. Hoofdstuk 1: pp. 9,12-14.
- [2] Sluis, A. van der; Vorst, H.A. van der; 1987. *Numerical solution of large, sparse, linear algebraic systems arising from tomographic problems*. Seismic tomography with applications in global seismology and exploration geophysics (editor: Nolet, Guust). D. Reidel Publishing Company, Dordrecht. Hoofdstuk 3: pp. 55,58,59.
- [3] Nolet, Guust, 1984. *Solving or resolving inadequate and noisy tomographic systems*. Journal of Computational Physics **61**, pp. 464, 465, 469-476.
- [4] Strang, Gilbert, 1988. *Linear algebra and its applications*. Harcourt Brace Jovanovich, San Diego, pp. 92-95, 154-157, 442-451.
- [5] Censor, Yair, 1990. *On variable block algebraic reconstruction techniques*. Mathematical Methods in Tomography, Springer Berlin, pp. 134-136,138.
- [6] Björk, Åke; Elfving, Tommy; 1979. *Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations*. BIT Numerical Mathematics, nr. 19, pp. 148,149.



Ehrhart Theory applied to Voting Theory

Frank Tabak

In a United States presidential election each state has a certain amount of electors. The candidate who obtains the most popular votes in a state will win all the electors for that state. The winner is the candidate with the most electors. In the presidential elections of 2000 George W. Bush defeated Al Gore with 271 electoral votes versus 266 for Al Gore. Al Gore, however, had 50,999,897 popular votes, 543,895 more than George W. Bush. One might therefore argue that Al Gore should have won the elections instead of George W. Bush. Elections should yield the most preferred candidate as the winner, but this is not always the case. This article will demonstrate that it is possible for two voting systems to yield different candidates as the winner. Furthermore, a method will be presented to compute the probability of this disagreement.

Voting Theory

The purpose of this article is to compute certain probabilities involved in elections. To be able to do this we must first model an election. The model requires us to make certain assumptions. In our case we have three assumptions:

1. The IAC condition is satisfied.
2. No strategic votes are cast.
3. Every voter will have a complete linear preference ranking on the candidates

IAC stands for *Impartial Anonymous Culture*. When the IAC condition is satisfied it is assumed that every voting situation, thus every outcome of an election, is evenly likely to occur.

The second assumption is that the voters will not vote strategically but sincere, that is, the voters will always vote for the candidate they prefer.

Finally, it is assumed that every voter will have a complete linear preference ranking on the candidates, thus a voter will always prefer one candidate over another and will not prefer them equally. The preference should also be *transitive*, that is, when one prefers candidate a over b and b over c then one automatically prefers a over c . This voter is said to have preference abc . The total amount of voters

with preference abc is denoted n_{abc} . The total number of voters is represented by the parameter N .

With these three assumptions we can model elections. An election is a decision-making process to *elect* a candidate or alternative. Elections use certain voting systems to make these decisions. The most common voting system is Plurality Voting, abbreviated to PV. In this voting system, a voter simply votes for a certain candidate and the candidate with the most votes wins the election. This voting system is widely used all over the world, like in the United States presidential elections. Plurality Runoff Voting (PRV) is a slightly different voting system. In this system a candidate should obtain the absolute majority of the votes, that is, more than 50%. If no one candidate meets that requirement, a second round of elections is held in which only the two candidates who obtained the most votes in the first round compete. This voting system is used for the French presidential elections, for instance.

It is possible that these two voting systems, PV and PRV, yield a different winner. Thus for a certain voting situation PV might yield candidate a as winner while candidate b would have won if PRV was used. Note that, for this to happen, no one candidate is to obtain the absolute majority of the votes in the first round. We can formulate this disagreement as a set of linear inequalities:

$$\begin{aligned} n_{abc} + n_{acb} &> n_{bac} + n_{bca} \\ n_{bac} + n_{bca} &> n_{cab} + n_{cba} \\ n_{abc} + n_{acb} + n_{cab} &< \frac{N}{2} \\ n_{abc} + n_{acb} + n_{bac} + n_{bca} + n_{cab} + n_{cba} &= N \\ n_i &\geq 0 \quad \text{for } i \in S_{abc} \end{aligned}$$

The first two inequalities say that in the first round a has the most votes, then b and c have the least votes. The third inequality represents the second round. In the second round candidate c does not participate anymore, which means that the voters with preference cab and cba will vote for their second choice. Therefore cab would vote for a , as will abc and acb , hence $n_{abc} + n_{acb} + n_{cab} < \frac{N}{2}$. This means that candidate a gets less than 50% of the votes, therefore candidate b has the absolute majority. The fourth inequality represents the total amount of voters. The last line makes sure that the number of people with a certain preference is not negative. S_{abc} is the set of all permutations of a, b and c .

Now that we have seen that a certain voting outcome can yield different winners when different voting systems are used, one might be curious as to what the



probability is that this event occurs. To do this we only have to count all the voting outcomes that will result in a disagreement and divide this number by the total number of voting outcomes. Note that the first assumption is used here, that is, that all voting outcomes are evenly likely to occur.

Ehrhart Theory

To count the number of voting outcomes that will result in a disagreement we need to use the Ehrhart Theory. The whole theory will not be presented here, only the results of the theory will be considered. For an extensive introduction to the Ehrhart Theory the reader is referred to [1].

To understand the results of the Ehrhart Theory the reader needs to be familiar with polyhedra. A *polyhedron* is described by the intersection of finitely many half-spaces. Recall that a half-space $\mathcal{H}$ is defined as the set

$$\mathcal{H} = \{\mathbf{x} \in \mathbb{R}^d : a_1x_1 + a_2x_2 + \dots + a_dx_d \leq b\}$$

with $a_1, a_2, \dots, a_d \in \mathbb{R}$ and not all $a_i = 0$ and $b \in \mathbb{R}^d$. Thus a polyhedron will be a set of $\mathbf{x} \in \mathbb{R}^d$ satisfying the system of m linear inequalities (with m the number of half-spaces):

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1d}x_d & \leq & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2d}x_d & \leq & b_2 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{md}x_d & \leq & b_m \end{array}$$

In Figure 1 an example of a 3-dimensional polyhedron is depicted.

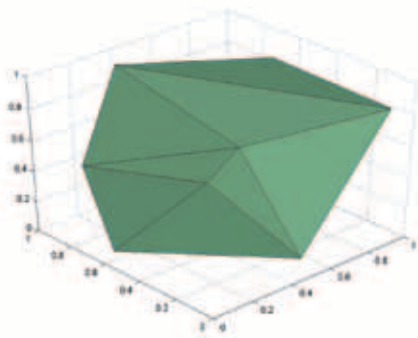


Figure 1: An example of a polyhedron

The problem considered earlier could also be represented by a polyhedron. This polyhedron, however, is dependent on a certain parameter N , the number of voters. Such polyhedra are called *parametrised polyhedra*.

A polyhedron can contain integer points, also called lattice points. The Ehrhart Theory is concerned with counting these points. Ehrhart's Theorem, the result of the Ehrhart Theory, states that the number of lattice points inside a d -dimensional polyhedron dependent on a parameter N is given by a polynomial in N of degree d . This polynomial is called the *Ehrhart polynomial*.

Barvinok's Algorithm

There are many ways to determine the Ehrhart polynomial of a parametrised polyhedron P_N . One could for example count the number of lattice points in P_1 (the polyhedron with $N = 1$), P_2 , ... by hand and use this information to determine the coefficients of the Ehrhart polynomial. This is the most straightforward method to determine the polynomial, it is, however, a tedious job to work like this.

Alexander Barvinok is a mathematician who developed an algorithm to count the integer points inside a polyhedron using the Ehrhart Theory. Explaining the whole algorithm is beyond the scope of this text, it can be found in [2, Sec. 3]. This algorithm is implemented by S. Verdoolaege, the implementation is called *barvinok*, not to be confused with the actual mathematician. Barvinok was used to obtain the results that will be presented later on in this text.

Counting Integer Points

To compute the probability of disagreement between PV and PRV for an election with 3 candidates and N voters, we have to count the number of election outcomes that will result in a disagreement, P_D and divide this by the total number of voting outcomes, P_T . We have seen that the problem could be represented by a polyhedron: the integer points in this polyhedron are all the voting outcomes in which disagreements occur. If we can determine the Ehrhart polynomial for this polyhedron, we can compute the probability of disagreement for all N . Note, however, that P_D are all the voting outcomes in which candidate a is the PV winner and b the PRV winner. There are 6 different pairs of PV and PRV winners thus:

$$\mathbb{P}(\text{PV and PRV disagree}) = \frac{6P_D}{P_T}$$

Computing the Probability

Using *barvinok* to determine the Ehrhart polynomial of our polyhedron, we can compute the probability of disagreement between PV and PRV for N voters. The results of these computations are listed below.

N	$\mathbb{P}(\text{PV and PRV disagree})$
10	0
15	0.0728
20	0.0434
30	0.0668
50	0.0839
100	0.1022
1000	0.1210
100000	0.1232
10000000	0.1233

It seems that the probability of disagreement converges to about 12% as $N \rightarrow \infty$. One can indeed show that this is true by analysing the Ehrhart polynomial.

Thus we can conclude that the probability that the two voting systems disagree is considerable. However, it is not possible to argue that PV is better than PRV or the other way around, both voting systems have their advantages and disadvantages.

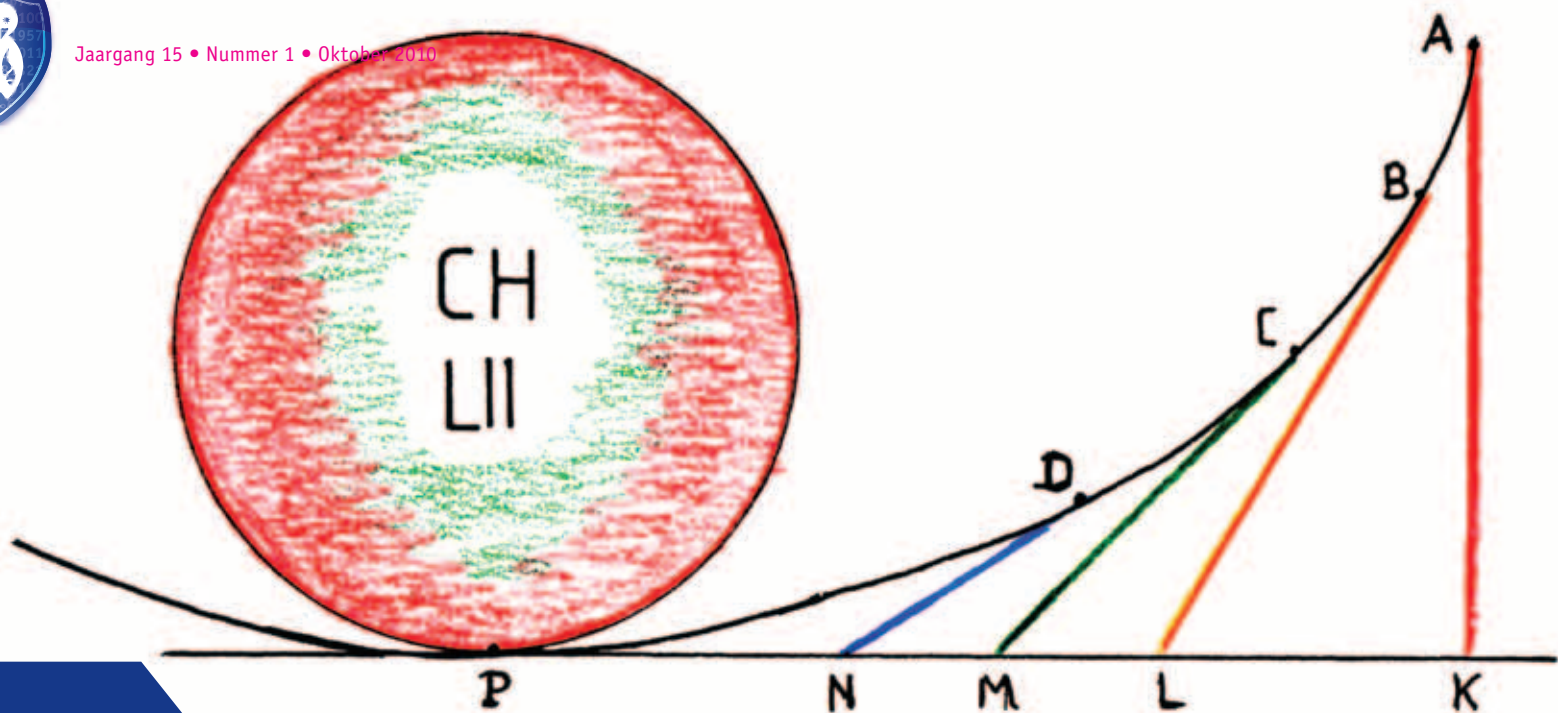
Other Voting Systems

The results obtained in this text were already known, but had been obtained by using different methods. These methods used tedious and complicated computations and reasonings. The Ehrhart Theory and Barvinok's algorithm have been in use for a long time in other areas of mathematics, and were only recently 'rediscovered' in the voting theory.

The Ehrhart Theory and Barvinok's algorithm can be applied to many other problems in the voting theory. One could, for instance, investigate the probability that an election is manipulated by a coalition of voters. Note that to do this, one should make different assumptions from the ones used here.

References

- [1] M. Beck and S. Robins, *Computing the Continuous Discretely: Integer-Point Enumeration in Polyhedra*, Springer, 2007.
- [2] S. Verdoolaege, R. Seghir, K. Beyls, V. Loechner, and M. Bruynooghe, *Counting integer points in parametric polytopes using barvinok's rational functions*, *Algorithmica*, 48:37-66, June 2007.



Wiskundepuzzel LII, the revenge!

Prelude tot de isochronie

Prof. dr. J.M. Aarts

Voor mijn verjaardag kreeg ik het boek dat hieronder zo fraai staat afgebeeld. Het gaat over geld dat moet rollen en het Twee-Euromuntje huppelt vrolijk over de kaft.

Zijn baan is een cycloïde, een rolkromme, hoe kan het ook anders. De rolkromme is uitvoerig bestudeerd door Christiaan Huygens. In de bovenstaande plaat is de cycloïde nogmaals getekend; je moet je daarbij voorstellen dat de speciale munt rolt langs een denkbeeldige horizontale lijn door A, als het ware langs het plafond. Het punt dat de cycloïde beschrijft is P. Zo met de bolle kant naar beneden herken je de cycloïde misschien als de halfpipe. In de guur heb ik de raaklijnen AK, BL, CM, DN getekend.

Als ik nu een munt langs een raaklijn laat rollen (of vallen) zonder rolweerstand. Dan vind ik voor iedere raaklijn dezelfde valtijd vanaf raakpunt tot aan het grondpunt, dus van A tot K, van B tot L, enzovoort...

Dit is de puzzel; een bewijs van deze uitspraak is goed voor drie punten op de ladder. Er zijn bonuspunten te verdienen: hoe meer meetkunde, des te beter! Ik zie de uitspraak als een voorloper van de isochronie van de cycloïde: als ik een munt (zonder rolweerstand) laat rollen langs de cycloïde dan is de tijd van heen en weer bewegen (de slingertijd) onafhankelijk van de plaats waar ik de munt op de cycloïde loslaat, en wel gelijk aan 2 keer de valtijd langs de raaklijn. En dat alles wordt door Christiaan Huygens bewezen zonder differentiaalrekening, alleen maar met gewone meetkunde.



Ladder

Ook al is de puzzel niet opgelost, de ladder gaat gewoon door! Als we alle correcties en rectificaties doorvoeren, komt het erop neer dat Marjon Ruijter vorig nummer de ladder gewonnen heeft en een prijs heeft verdiend. Thijmen Krebs heeft als enige de vorige puzzel opgelost en heeft daarmee een streepje voor op de anderen die ook 10 punten hadden. Dus hij heeft nu ook gewonnen en krijgt ook een prijs. Allebei worden ze uiteraard op 0 punten gezet. Gefeliciteerd Marjon en Thijmen!

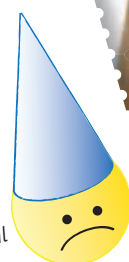
Op dit moment staat Misha Stassen samen met Josse van Dobben de Bruyn bovenaan de ladder met 10 punten!

Top 8

Misha Stassen	10
Josse van Dobben de Bruyn	10
Jeroen Wille	9
Jarno Hartog	9
Micke Vrolijk	6
Leon Planken	6
Jonatan Bijl	6
Charlotte Ipema	6

Het is jammer genoeg niemand gelukt om deze puzzel op te lossen! Vandaar dat we geen nieuwe puzzel hebben dit keer, maar iedereen nog een keer de kans geven om deze puzzel op te lossen.

Succes! En antwoorden kunnen voor 29 oktober naar machazine@ch.tudelft.nl gestuurd worden.



CAN YOU
READ THE
TEXT ON
THIS SCREEN

WHILST
SIMULTANEOUSLY
LOOKING
AT THIS ONE

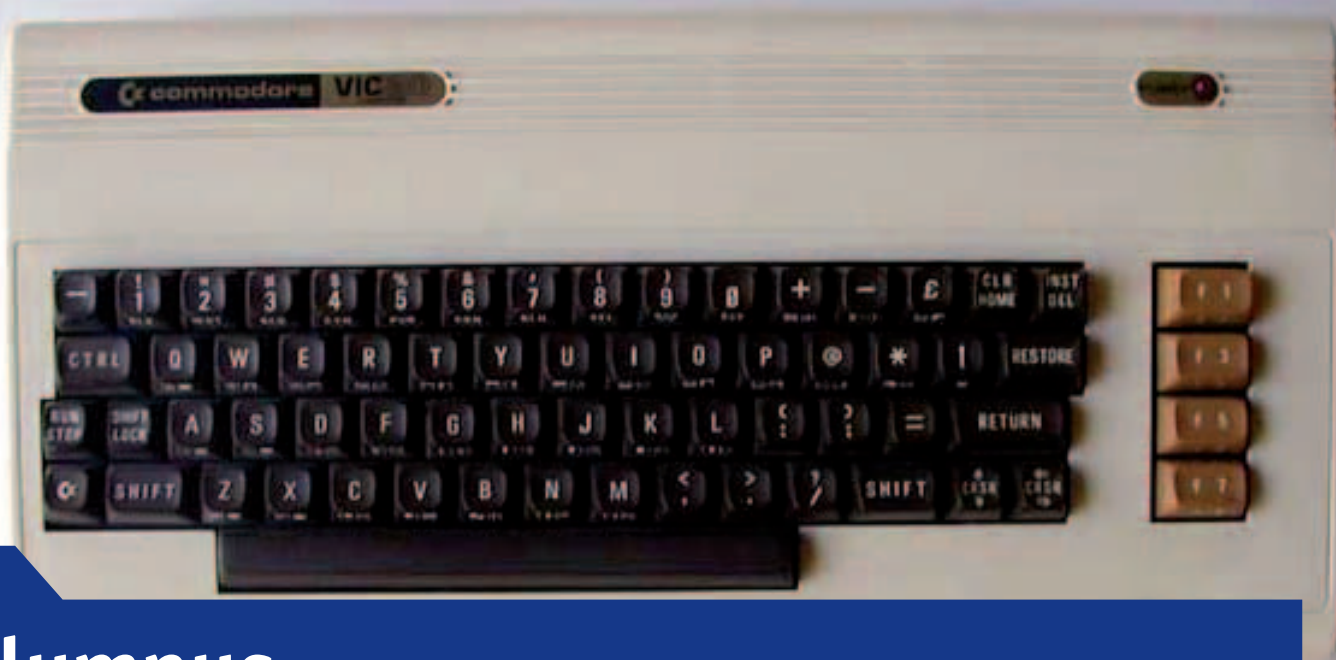
IN ADDITION TO
KEEPING
AN EYE ON
THIS SCREEN

AND WATCHING
THIS ONE
AS WELL?

Sign up now for the Traders Trophy on November 30 at Delft University.
Contact W.I.S.V. 'Christiaan Huygens' for more information.

WE ARE SCOUTING FOR **BRILLIANT** MINDS ONLY
START YOUR CAREER IN **TRADING** → APPLY AT WWW.OPTIVER.COM

optiver 



Alumnus Peter Disselkoen

In 1989 begon ik met mijn Informaticaopleiding. Mijn eerste keuze was gevallen op een studie economie in Rotterdam, maar toen bleek dat veel van mijn klasgenoten hier ook voor kozen, was het volgens de

economische wetten van vraag en aanbod onverstandig om economie te gaan studeren. Mijn tweede keuze was informatica aan de TU Delft.

Tijdens mijn keuze voor het middelbaar onderwijs had ik al informaticus als toekomstig beroep/carrière aangegeven, naast astronaut en piloot.

Mijn liefde voor computers begon al in mijn twaalfde levensjaar. Ik heb toen voor het eerst een programma geschreven. Dit was voor de Commodore VIC20, de voorloper van de Commodore 64. Het was een voorraad-administratiesysteem voor mijn vaders tuinbouwbedrijf. Hij raakte onder de indruk en is toen zelf ook gaan programmeren. Dit was een geluk voor mij, omdat er zo eerder geïnvesteerd werd in nieuwe hardware, zoals de Commodore 64 en een floppydiskdrive van 160 kB per floppy.

Tijdens mijn middelbare schooltijd heb ik veel geprogrammeerd en dit heeft mij later geholpen tijdens mijn studie. Programmeren vond ik altijd al leuker dan spelletjes spelen, iets wat andere klasgenoten veel deden. Mijn liefde voor wiskunde en exacte vakken heeft mij nog het meeste geholpen tijdens mijn studie. Als je hier geen gevoel voor had, kon je op de eerste dag al vertrekken. Dat was kort samengevat de eerste boodschap op een introductiecollege.

Studeren

De universiteit was een verademing. Je hoefde aan niemand verantwoording af te leggen. Of je wel of niet op college kwam, het maakte niets uit. De eerste weken kwam iedereen nog braaf op college, maar na een paar maanden was dat al een stuk minder. Dat gold voor mij ook.

Het eerste jaar heb ik weinig gestudeerd. Gelukkig kon ik nog goed weggkomen met mijn kennis, die ik had opgedaan tijdens het VWO. Maar de universitaire opleiding ging met een hoge snelheid door. Na één jaar had ik mijn propedeuse (P) nog niet gehaald. Nu kwam de grote uitdaging. Deze uitdaging heet

zelfdiscipline en doorzettingsvermogen, want zonder zelfdiscipline en doorzettingsvermogen red je het niet. Dit heb ik het tweede studiejaar geleerd. Ik heb later geleerd bij het aannemen van personeel dat deze twee eigenschappen vaak het verschil maken tussen de 'net' niet-afgestudeerden en de afgestudeerden.

En het was gelukt. Aan het einde van het tweede jaar had ik drie dingen bereikt, mijn P gehaald, zelfdiscipline en doorzettingsvermogen gekregen. Nu kwam mijn eigen trein ook goed op gang. Ik ging colleges volgen en studeren in de bibliotheek. De resultaten waren er ook naar. Deze gingen van zessen naar achten. Aan het einde van de studie werd nog voorgesteld om mij cum laude te laten afstuderen. Hiervoor moest nog wel een onvoldoende voor statistiek worden weg gewerkt. Hier had ik destijds weinig zin, dus dan maar niet. Achteraf gezien was het toch wel leuk geweest.

Werken tijdens de studie

Ondanks dat de studiebeurs hoger was dan tegenwoordig, was het geen vetpot. Om een leuk uitgaansleven te hebben en vakanties, was het belangrijk om genoeg bij te verdienen. Gelukkig kom ik oorspronkelijk uit het Westland. Door de vele tuinbouwbedrijven was daar altijd werk, waar je ook goed voor betaald werd. Maar het was beter om mijn kennis en kunde op IT vlak te gebruiken. Dat is een voordeel van informatica. Met een kleine investering in hardware kan je al bijverdienen.

Ik ben tijdens mijn studie software gaan schrijven voor exporteurs en transporteurs. Mijn oom had zo'n bedrijf en had veel problemen met de administratie van containers. Ik heb samen met hem en mijn neef een applicatie geschreven. Deze applicatie verkocht ik voor 1500 gulden. Nadat de applicatie klaar was, heb ik deze applicatie diverse malen verkocht tijdens mijn studie. Ik ging langs andere transporteurs en exporteurs en demonstreerde de applicatie. Vaak had één op de drie interesse. Nu werd het leven een stuk simpeler. Als het geld op was, dan ging ik weer een pakketje verkopen. Uiteindelijk is het zelfs vertaald naar het Italiaans.

Het ondernemen kwam hierdoor langzaam tot bloei. Het geluk was dat ik samen met een medestudent, Patrick van Dalen, tijdens het derdejaars practicum tegen een leuke opdracht aanliep. We deden het derdejaars practicum voor het Scheepvaart en transportcollege in Rotterdam. Daar maakten zij gebruik van een simulatiespel geschreven in Lotus 123 ter ondersteuning van hun



lesprogramma. Wij hebben toen samen een voorstel gemaakt om dit om te zetten in een mooie applicatie. Uiteindelijk hebben wij voor hen meerdere applicaties mogen maken.

Dit betekende het begin van het ondernemerschap. In 1995, na onze studie, hebben we Pulsar Software V.O.F. opgericht. Het eerste jaar na onze studie was zwaar. We verdienden weinig en hadden ons kantoor in het 'washok' van Patricks studentenhuus. Wanneer ik nu terugkijk, vroegen wij te weinig voor onze diensten. Patrick had bijvoorbeeld een opdracht bij Philips Medical Systems. Hiervoor vroegen wij ongeveer 40 gulden per uur. Wij vonden dat toen veel, maar achteraf gezien had het minstens 100 gulden mogen zijn. Zeker gezien de besparingen die wij in korte tijd voor hen hadden gerealiseerd.

Na veel fouten ging het tweede jaar een stuk beter. Het eerste jaar als ondernemer verdienden wij elk 1000 gulden per maand. Oud-medestudenten kwamen allemaal even langs met hun lease auto's, terwijl wij alleen een tweedehands auto hadden. Het tweede jaar verdienden wij elk 7.500 gulden per maand.

Nu was de stap aangebroken om te professionaliseren. Pulsar Software V.O.F. is omgezet in Vixion Software BV. We zijn verhuisd naar Radex. Dit is een bedrijfsverzamelgebouw tegenover de Lucht- en Ruimtevaartfaculteit. Mijn opa heeft ons hierin gesteund door middel van een achtergestelde lening voor de huur. Hij vond het 'washok' niet professioneel en dat had hij goed gezien.

Door Radex zijn we in contact gekomen met veel kleine startende bedrijven, die een TU achtergrond hadden. Via deze andere contacten hebben wij opdrachten voor grotere partijen kunnen uitvoeren, waaronder Delta Lloyd en KPN Mobiel. Nu was de tijd aangebroken om te gaan groeien. We zijn gegroeid van 2 naar 7 medewerkers in twee jaar.



Eind 1999 deed zich een kans voor, dachten wij. Wij konden Vixion Software inbrengen in Tjip BV in ruil voor aandelen. Tjip is een IT bedrijf dat zich toelegt op het ontwikkelen van software voor de financiële markt. Bij Tjip was ik een medewerker en niet meer een ondernemer. Iets wat mij na 3 maanden al tegen begon te staan. Ik heb het uiteindelijk 3 jaar uitgehouden. Wel heb ik veel geleerd hoe je een bedrijf moet managen en laten groeien. Verder heb ik veel contacten opgedaan in de financiële wereld.

Na Tjip moest ik weer helemaal opnieuw beginnen. Het eerste jaar heb ik samen met een vriend geprobeerd de beurs (Eurex) te kraken en diverse simulatoren en automatische handelssystemen geschreven om de beurs te verslaan. Soms ging het erg goed, maar na één jaar was het geld op. Veelal veroorzaakt door kleine domme fouten, maar elke fout(je) wordt met keiharde euro's betaald. We zijn zelfs door de Deutsche Börse benaderd om als broker te gaan optreden, omdat we zoveel handel deden (soms rond de 2% van de daghandel). Ik ben nog steeds van mening dat het systeem te verslaan is, maar snelheid is van belang. Uit de simulator bleek dat met zeer snel op beurskoersen reageren er geld verdiend kon worden. Maar 0,8 seconden was nog steeds te langzaam. Hiervoor moet je systemen op de beurs zelf hebben, die direct zijn aangesloten op het netwerk van de beurs. Dit is een trend die momenteel door grote handelshuizen en beurzen is ingezet.

Na onze ervaring op de beurs was het weer tijd om betaalde opdrachten te gaan doen. De eerste opdracht kreeg ik in 2004. In één maand had ik voor een verzekeraar een applicatie geschreven, die het werk van 20 uitzendkrachten overbodig maakte. Een directeur was hiervan zo onder de indruk, dat er gelijk diverse vervolgoopdrachten aankwamen. Dit was de basis voor een nieuw bedrijf.

In 2006 is Vixion BV opgericht. Vixion heeft zich momenteel toegelegd op het ontwikkelen van complexe rekenboxen voor de financiële wereld. Actuariële medewerkers van grote maatschappijen kunnen producten definiëren in Excel, die daarna worden omgezet in een snelle webservice. Het product wordt hierdoor ontsloten voor andere applicaties. Het meest complexe product tot nog toe is een adviesapplicatie voor collectieve pensioenen. Hiermee kunnen verschillende pensioenproducten van verschillende aanbieders worden vergeleken. Zonder goede rekenondersteuning is dit anders niet te doen. De rekenbox rekent hierbij elke medewerker van een bedrijf door. Per medewerker moeten ongeveer 55.000 formules worden doorgerekend. Snelheid is hierbij van groot belang. Momenteel gebeurt dit in 350 milliseconden, maar we willen terug naar 200 milliseconden.

Ik heb wel gemerkt dat de vele wiskunde vakken tijdens de opleiding nu heel goed van pas komen. Verder merk ik, dat de opgebouwde kennis van IT tijdens mijn studie na 15 jaar nog steeds goed te gebruiken is. Natuurlijk zijn er ook nieuwe ontwikkelingen gekomen die belangrijk zijn, zoals cloud computing. Maar in de basis is het nog steeds hetzelfde: invoer – berekenen – uitvoer. Door de toenemende rekenkracht heeft de mens de mogelijkheid om de wereld complexer te maken en dat zijn we met zijn allen aan het doen. Mijn uitdaging ligt erin om het met behulp van IT weer eenvoudiger te maken, maar dat is niet altijd even eenvoudig.

Voor vragen en of commentaar ben ik bereikbaar per e-mail op Peter.disselkoen@vixion.nl.

Met vriendelijk groet,

ir. P.L. Disselkoen. 

Historisch Persoon

Gottfried Wilhelm Leibniz

Michiel van Dam

Gottfried Wilhelm (von) Leibniz was een veelzijdig Duits wiskundige, filosoof, logicus, natuurkundige, historicus, rechtsgeleerde en diplomaat en wordt beschouwd als een van de grootste denkers van de 17e eeuw.

Biografie

Jeugd en studiejaren

Leibniz werd op 1 juli 1646 geboren in Leipzig. Leibniz was nog maar zes toen zijn vader overleed, maar op die leeftijd had hij al een passie ontwikkeld voor lezen en studeren. Hij leerde Latijn en bestudeerde zijn vaders bibliotheek, die vol stond met Latijnse klassiekers, filosofische en religieuze werken.

Toen Leibniz 15 was, betrad hij de universiteit van Leipzig om filosofie te studeren. Twee jaar later maakte hij kennis met elementaire algebra en Euclidische meetkunde op de universiteit van Jena. Hier begon hij zijn ideeën van een universeel 'alfabet van de menselijke gedachten' te ontwikkelen. Hierin probeerde hij menselijke gedachten vorm te geven door middel van een voor iedereen begrijpelijke tekentaal van symbolen.

Hij haalde zijn graad in 1664. Zijn dissertatie voor de graad van doctor in de rechten werd geweigerd. De reden daarvoor was waarschijnlijk zijn leeftijd, maar misschien waren er ook politieke problemen. Hierom verliet hij Leipzig en kreeg hij deze graad op twintigjarige leeftijd aan de universiteit van Altdorf in Neurenberg.



Carrière

Leibniz is begonnen als alchemist in Neurenberg, hoewel hij niets over dit onderwerp wist. Al snel ontmoette hij echter Johann Christian von Boineburg, de toen tijdelijk ontslagen eerste minister van de keurvorst van Mainz. Von Boineburg huurde Leibniz in als zijn assistent, en toen hij zich kort daarna verzoende met de Keurvorst, introduceerde hij ook Leibniz bij hem. Leibniz wijdde een van zijn essays over het recht aan de keurvorst, dit in de hoop zijn kans op werk aan diens hof zo te vergroten. Deze strategie werkte zoals gepland; de keurvorst vroeg Leibniz te assisteren bij de herziening van het

Wetboek voor het keurvorstendom Mainz. In 1669 werd Leibniz benoemd tot assessor bij het Hof van beroep. Nadat von Boineburg in 1672 overleed, bleef Leibniz nog tot 1674 in dienst van diens weduwe.

Doordat hij bij Van Boineburg in dienst was, vielen Leibniz zijn nota's en brieven op en stelden hem in een gunstig daglicht. Dit maakte een diplomatieke rol voor hem mogelijk. Leibniz publiceerde een essay onder een pseudoniem, waar hij zich hard maakte voor de Duitse kandidaat voor de Poolse kroon. De realiteit was echter dat, gedurende het volwassen leven van Leibniz, Lodewijk XIV van Frankrijk de bepalende ambities had. Deze vorst had serieuze plannen om Frankrijk uit te breiden tot aan de Rijn. De Dertigjarige Oorlog had het Duitstalige deel van Europa al ontvolkt en verder gefragmenteerd. Leibniz stelde voor om Duitstalig Europa te beschermen door Louis als volgt af te leiden. Frankrijk zou worden uitgenodigd om Egypte te veroveren als een opstap naar een uiteindelijke verovering van Nederlands-Indië. In ruil zou Frankrijk er mee instemmen om het Duitse rijk en de Nederlanden met rust te laten. Dit plan kreeg voorzichtige steun van de keurvorst van Mainz. In 1672 verzocht de Franse regering Leibniz naar Parijs te komen voor verdere discussie over het plan, maar het werd al snel ingehaald door de realiteit van het uitbreken van de Hollandse Oorlog en werd irrelevant.

Christiaan Huygens

Zo begin Leibniz zijn Parijse periode. Kort na zijn aankomst ontmoette hij Christiaan Huygens. Door dit contact en zijn iets latere ervaringen in Engeland kwam hij tot het inzicht dat zijn kennis van de wis- en natuurkunde nog te fragmentarisch was. Met Huygens als mentor begon hij een programma van zelfstudie dat binnen de kortste keren vrucht droeg. Al spoedig bleek hij in staat om belangrijke bijdrage aan zowel de natuur- als de wiskunde te leveren. In deze periode legde hij de basis voor zijn onafhankelijk van Newton geformuleerde versie van de differentiaal- en integraalrekening. Ook bestudeerde hij dankzij zijn contact met Huygens de zowel de gepubliceerde- als de ongepubliceerde geschriften van René Descartes en Blaise Pascal.

Nadat Leibniz zijn rekenmachine, de eerste rekenmachine, die alle vier de elementaire rekenkundige bewerkingen aankon, aan de Royal Society had gedemonstreerd, werd hij tot extern lid van de Royal Society gemaakt. In 1623 had Wilhelm Schickard al de eerste mechanische rekenmachine gebouwd, maar hiermee kon je alleen optellen en aftrekken.

Zeven fundamentele principes

Leibniz maakte in zijn filosofische werken gebruik van zeven fundamentele principes. Hij hield af en toe een betoog voor een van deze principes, maar vaker nam hij ze als vanzelfsprekend aan.

1. Als een uitspraak waar is, is de negatie van die uitspraak onwaar, en omgekeerd.
2. Twee zaken zijn identiek dan en slechts dan als ze precies de zelfde eigenschappen hebben.
3. Er moet genoeg reden (vaak alleen bij God bekend) zijn om iets te laten bestaan, om iets te laten gebeuren, of om nieuwe waarheden te ontdekken.
4. Er vindt geen interactie plaats tussen lichaam en geest, maar hun gedragingen lijken sterk op elkaar.
5. De natuur is continu.
6. God kiest altijd de beste optie
7. Door (6) is de actuele wereld het optimum van alle mogelijke werelden.



Toen twee van Leibniz beschermheren in dezelfde winter overleden, moest Leibniz een nieuwe basis voor zijn carrière vinden. In dit verband bleek een uitnodiging uit 1669 van de hertog van Brunswijk om een bezoek aan Hannover te brengen beslissend. Leibniz bedankte toen voor de uitnodiging, maar begon in 1671 wel een correspondentie met de hertog. In 1673 bood hertog Johan Frederik van Brunswijk-Lüneburg hem een post als adviseur aan, een aanbod dat Leibniz twee jaar later schoorvoetend aanvaardde, nadat hem duidelijk was geworden dat een verder verblijf in Parijs er op dat moment niet inzat.

Leibniz streefde zijn hele leven naar harmonie op zoveel mogelijk fronten. Behalve zijn monadenleer en zijn tekentaal trachtte hij tevens zoveel mogelijk wetenschappers tot samenwerking te bewegen. Hij stichtte indirect de Berlijnse Academie van Wetenschappen en ondernam ook pogingen om alle Christelijke kerken nader tot elkaar te brengen. Leibniz overleed op 21 juni 1716 in Hannover, waar hij tot zijn dood bibliothecaris was.

Leibniz als wiskundige

Logica

Leibniz is een grondlegger van het logisch formalisme. Hij kende als geen ander de effectiviteit van de objectieve wiskunde en hij probeerde een manier te vinden om de wiskundige methodiek te gebruiken om conflicten te beslechten. Hij vatte een redenering of argumentatie op als een berekening. Hij ontwierp hiervoor een objectieve symbolische taal waarin je een redenering kon opzetten. De gewone natuurlijke taal was ongeschikt doordat deze niet exact is, het kan door meerdere mensen anders geïnterpreteerd worden. Leibniz' ideeën waren vooruitstrevend voor zijn tijd, maar zijn logische formalisme was nog verre van compleet. Pas een kleine tweehonderd jaar later ontwierp George Boole het eerste volledige logische formalisme voor de propositiële logica.

Analyse

In 1663, terwijl Leibniz in Parijs was, ontwikkelde hij zijn algemene methode om hellingen te berekenen. De drie jaren daarop maakte Leibniz een enorme wiskundige ontwikkeling door en ontdekte hij fundamentele principes van de calculus. Zijn resultaten op het gebied van sommen en verschillen waren echter niet nieuw. Veel van zijn kennis was door zelfstudie verkregen, waardoor hij vaak al bestaande wiskunde herontdekte. Het originele van zijn werk was dat hij sommen en verschillen bij krommes onderzocht. Hij legde een kromme uit als een veelhoek met oneindig veel zijden. De verschillen die hij onderzocht werden hierbij oneindig klein, de differentiaal, of 'infinitesimalen'.

Leibniz ontwikkelde de calculus vanuit het idee dat som en verschil tegen-gestelde operaties zijn. Daarom is voor hem de geldigheid van de hoofdstelling van de integraalrekening (die het verband tussen de begrippen afgeleide en integraal geeft) ook vanzelfsprekend – daar is het allemaal mee begonnen.

Toen Leibniz zijn ontwikkelingen publiceerde, was hij ongerust over zijn gebruik van infinitesimalen. Dit begrip was nog niet goed gedefinieerd en hij wist dat er veel kritiek op zou komen. Daarom had hij het in zijn eerste publicatie over dx als een willekeurig eindig lijnstuk. Hij publiceerde zijn werk in 1684 in *Acta Eruditorum*, een wetenschappelijk tijdschrift dat hij zelf had opgericht, in een artikel getiteld 'Nova Methodus pro Maximis et Minimis, itemque tangentibus, qua nec fractas, nec irrationales quantitates moratur, et singulare pro illi calculi genus', oftewel 'Nieuwe methode voor maxima, minima en raaklijnen die niet gehinderd wordt door breuken of irrationale grootheden en een bijzondere rekenmethode daarvoor'.

Leibniz als filosoof

De filosofie van Leibniz lijkt soms gefragmenteerd, aangezien zijn filosofische stukken vaak een groot aantal kleine stukjes betroffen. Artikelen in tijdschriften, manuscripten die pas na zijn dood gepubliceerd werden, en veel brieven naar correspondenten. In tegenstelling tot Descartes en Spinoza, die in dezelfde tijd leefden, had Leibniz een goede universitaire opleiding filosofie gevolgd. Leibniz keek geïnteresseerd naar de methoden en theorieën van Descartes, Huygens, Newton en Boyle, maar bekeek ze wel allemaal door een bril van oude Aristotelianse scholastiek. Desondanks bleken de methoden en thema's die Leibniz behandelde weer terug te komen in de filosofie van de 20e eeuw.

Zijn leven lang zocht Leibniz naar een allesomvattende samenstelling voor wetenschap en filosofie. In 1714 publiceerde hij zijn werk *La Monadologie*, waarin hij zei dat alles bestaat uit ontelbare eenheden of krachtpunten van verschillende bewustzijnsgraad die hij monaden noemde, de individuele eigenschappen die het verleden, heden en toekomst van elk ding zouden bepalen. Hoewel de monaden onafhankelijk van elkaar waren, was hun interactie voorspelbaar. Voor hem betekende dat dat Christelijk geloof en wetenschappelijke redenering niet met elkaar in tegenspraak hoeven te zijn. Hij ging uit van een tevoren door God vastgelegde orde en gaf dit principe de naam *harmonia praestabilita*, waarmee zou zijn aangetoond dat de wereld zoals die is, de best mogelijke van alle werelden is.

Leibniz als informaticus

Leibniz kan beschouwd worden als de eerste informaticus en informatiekundige. Al vroeg in zijn leven beschreef hij het binaire talstelsel dat tegenwoordig voor computers gebruikt wordt. Hij voorzag algoritmische informatiekunde en enkele theorieën die Lagrange honderd jaar later zou ontwikkelen. Zijn rekenmachine bevatte al aspecten van de universele Turing machine en hij greep al naar hard- en softwareconcepten die pas later door Charles Babbage en Ada Lovelace uitgewerkt zouden worden. Toen hij in 1679 zijn binaire berekeningen besprak, stelde hij zich een machine voor waarin knikkers de binaire getallen voorstelden, beheerd door basale ponskaarten. In moderne computers zijn de knikkers en de zwaartekracht vervangen door registers, spanningsniveaus en elektronen, maar voor de rest werken ze nog steeds hetzelfde als Leibniz zich voorstelde.

Reputatie voor en na zijn dood

De reputatie van Leibniz werd al steeds minder voor hij overleed, zodanig zelfs dat geen van de organisaties waar hij bij aangesloten was melding maakten van zijn overlijden. Het enige boek wat van hem herinnerd werd was *Théodicée*, wat in een boek van Voltaire getorpedeerd werd. De voorstelling van zaken zoals Voltaire die presenteerde werd door veel mensen geloofd, wat de reputatie van Leibniz alleen maar slechter maakte. Ook twijfelde veel mensen in Europa eraan of Leibniz wel onafhankelijk van Newton zijn calculus had ontwikkeld, met vermoedens van plagiaat. Ook hier droeg Voltaire aan bij.

De opmars naar de status die Leibniz vandaag de dag wordt toegekend begon toen er meer werken van hem gepubliceerd werden. Immanuel Kant las zijn werken aandachtig en er volgden in de 19e eeuw herpublicaties van geschriften van Leibniz. In 1900 publiceerde Bertrand Russel een studie naar enkele werken van Leibniz, en daarna werden nog meer ongepubliceerde werken van Leibniz alsnog gepubliceerd, na redactie door Louis Couturat. Al deze publicaties lieten de reputatie van Leibniz stijgen tot het niveau waar het nu is, als een van de grootste denkers van de 17e eeuw. 

Historisch MaCHazine

Kees Boon

Zoals bovenaan de pagina te zien valt, is dit het eerste nummer van de vijftiende jaargang van het MaCHazine. Omdat vijftien te factoriseren is in vijf en drie en omdat er vijf jaren in een lustrum zitten, besloten we bij de redactie om dit jaar in de geschiedenis van het MaCHazine te duiken.

Vijftien jaar geschiedenis is een lange tijd, dus om daar wat grip op te krijgen was het zaak om de oude MaCHazines eens te bekijken. Deze taak werd aan mij opgelegd. Vol goede moed begaf ik mij naar de stoffige ruimte van het archief. Om te beginnen plande ik eerst maar eens alle MaCHazine's van de afgelopen veertien jaar bij elkaar te zoeken, zodat ik daarna chronologisch op zoek kon gaan naar historische artikelen.

Nu bleek dit makkelijker gezegd dan gedaan. Het archief is geen netjes geordende kast met een indexering erbij. Het woord 'puinhoop' beschrijft de situatie beter. Slechts een kast met het nummer zestien gaf aan waar de MaCHazine gerelateerde dingen opgeslagen waren. Ook deze nummering bleek niet meer dan een suggestie, gezien er tientallen ongemarkeerde dozen met MaCHazines door de gehele ruimte verspreid waren. Er stond me bij dat sommige van de oudere MaCHazines geheel verloren zijn gegaan. Dit werd nog een hele opgave.

Onvervaard ging ik aan de slag. Als een Indiana Jones (maar dan zonder hoed, zweep en special effects) duik ik de stoffige krochten van EWI in. Op zo'n onderzoekstocht vind je nogal wat historische CH artefacten uit een recent of ver verleden; een gigantische poster van president Obama, een Delfts blauw servies waarop de Study Tour naar Japan in 1985 vermeld staat, notulen van MaCHazine vergaderingen uit het jaar 2000 welke iemand kennelijk belangrijk genoeg vond om te bewaren, het programma voor het 6e CH lustrum (1987) en een bak 5 ¼ inch floppy's (1.2 MB volgens Wikipedia). Een indringend gevoel van de lange geschiedenis van onze studievereniging komt over mij heen.

Na enige tijd zoeken heb ik ongeveer een tijdlijn van de MaCHazines tot nu toe voor me op de grond liggen. Sommigen ontbreken nog, maar gek genoeg zijn dit alleen de meer recente nummers. Een enkele dreigt uit elkaar te vallen. Al met al ligt er een indrukwekkende rij kwartaalbladen. Hoewel, de eerste twee jaargangen verscheen het MaCHazine nog maar twee keer per jaar. De eerdere jaargangen hebben een A5 pagina formaat maar over je jaren heen evolueert het langzaam tot in de huidige afmetingen via B5 naar A4.

Nu de oude MaCHazines veilig gesteld waren, ging ik aan de slag om leuke oude artikelen te vinden. Met mijn pas opgedane historische en nostalgische gevoeligheid moest dit toch geen probleem zijn. De mij ver voorgaande redacties moeten een neus voor belangrijke gebeurtenissen gehad hebben, want al heel snel vind ik twee artikelen die verband houden met belangrijke historische gegevens. Ten eerste een artikel over de bekendste mijlpaal in de kunstmatige intelligentie: schaakcomputer Deep Blue die grootmeester Kasparov in 1997 voor het eerst verslaat. In het allereerste MaCHazine staat hier een artikel over. Ten tweede een stuk uit het eerste nummer van de tweede

jaargang over het (toen nog aankomende) millenniumprobleem. In 1999 was ik elf, dus ik herinner me alleen dat er nogal wat overtrokken reacties waren op Y2K. Kennelijk had het MaCHazine al meteen genoeg kwaliteit om er een net, niet paranoïde artikel over te plaatsen. Deze artikelen zijn hieronder en hiernaast opnieuw geplaatst.

Een kleine twee uur na aanvang kom ik moe en onder het stof weer uit de archieven vandaan met een doos vol MaCHazines. Na me enigszins gefatsoeneerd te hebben, loop ik blij weer terug naar het CH hok. Aldaar aangekomen kom ik interim-hoofredacteur Pul tegen. "Hoi Pul, ik heb alle MaCHazines tot nu toe die ik kon vinden uit het archief gevist." Pul: "Uit het archief? Wist je dan niet dat het bestuur gewoon alle nummers in de kast heeft zitten?"

Deep Blue einde van het schaken?

'Dus computers zijn nu intelligenter dan de mens?' was de onvermijdelijke vraag van de journalisten op de avond nadat Gary Kasparov in de schaakmatch tegen IBM's Deep Blue het verwilderde hoofd in de schoot had gelegd. Je kunt dat zo'n verslaggever niet echt kwalijk nemen. Je wilt immers dicht bij het volk blijven, en dan moet je meer Story in je stijl strooien dan Kijk. Hij stelde zijn vraag aan een echte IBM-deskundige (bril, blazer, betrouwbaar kapsel), die, leunend op het kastje waarmee het allemaal gebeurde was, schaaachtig glimlachend wat onduidelijk mompelde. Daardoor aangemoedigd vroeg hij het ook nog aan een echte schaker, Hans Böhm, en die dacht, al zei hij het niet met zoveel woorden, eigenlijk van wel. 'Tja', zei hij, 'de mens, die kan zich maar beter gaan richten op die dingen waar hij beter in is dan de computer, en dan kom je toch al gauw uit bij liefde en begrip en zo'. Daarmee had het journaal twee mythen opnieuw onuitroeibaar in de volksziel verankerd: dat een computer een kastje vol draaden is, en dat computers intelligent zijn.

Maar het had er tevens één genadeloos onmaskerd: van schaken word je niet intelligenter.

Dat laatste geldt evengoed voor computers als voor mensen. Deep Blue doet niets anders dan de eerste de beste Nintende slaafs volgens voorschrift de binnenkomende gegevens bewerken, en het resultaat daarvan uitspugen. Pulsjes in, pulsjes uit, anders niks. Dat er ergens aan een schaaktafelje iemand zijn haren uit zijn hoofd zit te trekken laat het ding ijskoud. Knappe of onzinnige zetten maken niet uit, Deep Blue gaat ze allemaal met dezelfde ijver te lijf. Hoogstens zal het rekenwerk bij een rare zet iets langer duren, omdat er meer voortzettingen die eerder al als onwaarschijnlijk terzijde waren geschoven, toch moeten worden doorgetrokken. Je kunt dan roepen dat de computer 'verraat' is, maar dat is hetzelfde als roepen dat je auto verraat is als je plotseling zo hard aan het stuur ruk dat hij in een slip raakt. Dat juist het schaakspel vanaf de beginjaren van de computer een geliefd doelwit van fabrikanten van



hardware en software is geweest, is geen toeval. Aan de ene kant is schaken aantrekkelijk vanwege de mysterieusheid ervan. Als er één groep is die zijn PR goed gedaan heeft, dan zijn het wel de verzamelaars schaakfanaten van deze wereld. Schaken heeft een welhaast onaantastbaar prestige. Het is niet alleen het meest 'salonfähige' spel dat er bestaat, er worden ook wonderbaarlijke heilframe eigenschappen aan toegedicht. Vooraanstaande mensen binnen de schaakbond willen schaken zelfs als schoolvak invoeren. Wie dus een computer kan leren schaken, is verzekerd van de peilloze bewondering van vrijwel iedereen. En wie zegt daar nu nee tegen?

Tegelijkertijd is schaken niets anders dan een ingewikkelde vorm van boter, kaas en eieren. Er is, net als bij dat flauwe spelletje, een eindig aantal velden (65: 32 witte, 32 zwarte, en 1 'van het bord af'), en

Net als bij boter, kaas en eieren zou je in principe alle paden, alle opeenvolgende toeken bordstanden, kunnen uitrekenen die eindigen met een gewonnen of remisestand. Als je de circulariteiten uitsluit, is ook dat aantal paden eindig, evenals hun lengte. Je kunt het schaakspel dus compleet uitrekenen. Mocht daarbij blijken dat er, net als bij boter, kaas en eieren, paden bestaan die, ongeacht wat de tegenstander doet, altijd in een gewonnen stand eindigen, dan is het schaakspel dood. Wie wint dan altijd, met hetzelfde gemak waarmee wij kleine netjes met kruisjes en cirkeltjes verpletteren.

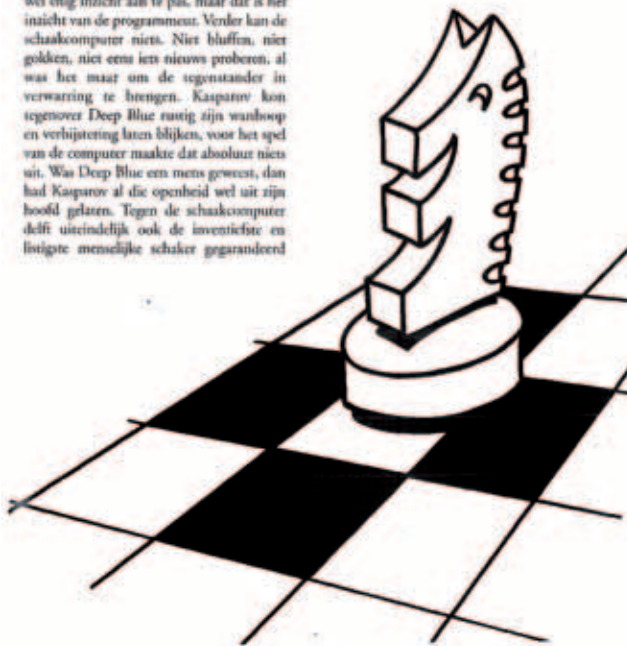
Grote aantallen gegevens doorrekenen en onthouden, dat is precies waar computers uitstekend in zijn, en mensen juist slecht. Dat verschil maakt schaken voor zowel schakers als programmeurs zo aantrekkelijk. Juist omdat het aantal paden veel groter is dan wij kunnen onthouden, komt aan schaken intelligent gedrag te pas: inschatten wat op een bepaald moment de kansrijke voortzettingen zijn zonder de hele route te kennen, in de gedachten van de tegenstander, zijn plannen, zijn zwakke plekken en zijn irritaties herkennen. Omgekeerd kan een beter met paden gevuld geheugen een gebrek aan intelligent gedrag meer dan goedmaken. Beroepsschakers studeren zich niet voor niets een rotje op oude partijen. Sterker nog, binnen een gesloten en berekenbaar systeem als schaken wint geheugen het uiteindelijk altijd van intelligentie. Intelligentie speelt met kansen, geheugen met zekerheden.

Dat is ook de reden dat schaakcomputers steeds beter worden. De snelle

machines en slim ontworpen programma's van nu kunnen binnen redelijke tijden zoveel voortzettingen, zoveel dieper doorrekenen dan zelfs de beste menselijke tegenstander, dat het gemis aan intelligentie ruimschoots gecompenseerd wordt. De schaakcomputer met een uitputtende lijst van paden en bordstanden bestaat nog niet, maar wel kunnen ze door hun rekenkracht al beter de kansrijke voortzetting kiezen. Daar komt uiteraard wel enig inzicht aan te pas, maar dat is het inzicht van de programmeur. Verder kan de schaakcomputer niets. Niet bluffen, niet gokken, niet eens iets nieuws proberen, al was het maar om de tegenstander in verwarring te brengen. Kasparov kon tegenover Deep Blue rustig zijn wanhoop en verbijstering laten blijken, voor het spel van de computer maakte dat absoluut niets uit. Was Deep Blue een mens geweest, dan had Kasparov al die openheid wel uit zijn hoofd gelaten. Tegen de schaakcomputer delft uiteindelijk ook de inventiefste en listigste menselijke schaker gegarandeerd

het onderlip. Dat zegt niets over de intelligentie van mens of computer, de computer zal zelfs beter gaan schaken naarmate zijn programma suffer kan worden omdat zijn geheugen voller raakt. Echt schaken blijft iets voor mensen onder elkaar. Wie dat niet genoeg vindt, kan zich altijd nog op liefde en begrip storten. Daar is nog wel behoefte aan.

overgenomen uit het NRC Handelsblad van 17 mei 1997



Beroepsschakers studeren zich niet voor niets een rotje op oude partijen

een eindig aantal stukken, die elk vanaf elk veld een eindig aantal bewegingen kunnen uitvoeren. Er is dus een eindig aantal mogelijke bordstanden, waarvan sommige 'gewonnen' betekenen, andere 'remise', en weer andere 'kan nog van alles worden'.

The Year 2000 and Embedded Systems:

For Most Businesses, This Does Not Have to be a Major Problem

overgenomen van Internet - Year 2000 Information Center™

One second after midnight December 31st, 1999 most of the elevators in all our cities go down into the basements of their buildings and shut themselves off. Heating and air conditioning units fail because the thermostats no longer function. Doors that open with electronic "card" keys stay locked, phones are out, bank vault doors won't open or open by themselves at the wrong times. Fax machines quit, cars run badly or not at all, traffic lights won't cycle correctly and airliners get lost flying across the country. Could this really happen? Not really, unless everybody sticks their head in the sand and does nothing for the next two years, pretending nothing could go wrong. The problems above are caused by embedded chips in devices and systems in non-computer applications, many not that obvious. We have become increasingly dependent on these chips which reside in everything from our thermostats and phones to the smoke detectors on our ceilings. They are everywhere and the prospect of some number of them just

quitting at the century change is really scary. But we do have two years to fix them and there aren't that many of them, really. Depending on the size of the business, there may be several hundred systems and machines, and maybe thousands. But this isn't like having fifty-five million lines of code that need to be rewritten and tested. The reason it is not going to be a huge problem is that few of the machines and systems that can fail are critical to the success of the business.

Almost everything has a chip in it anymore. I remember a very old Saturday Night Live show where one of the running jokes was how everything was becoming digital. One of the sight gags was becoming digital. I'm sure if there is anything down there we want monitored, someone will come up with boxes with a chip! We can even inject chips in our pets and livestock to identify and protect them from theft. There really are chips in almost everything. Those chips were programmed at the factory to the customer specifications and nobody knows if they will work past the year 2000 or not.

We could have some real problems here, especially with machines and systems that either sustain life, such as machines in hospitals, or where lives are depending on the information from a chip, such as an airplane's instruments. Chips that automatically monitor a process like a production line or the decay of nuclear materials in a storage facility are also very important for obvious reasons.

Most of the systems that use embedded chips don't use dates and most of them aren't in places that will cause life or wallet threatening problems if they fail. We have all seen VCRs at the office blinking 12:00 all the time. They aren't daily or weekly compliant, let alone 2000 compliant, but nobody cares. It just doesn't matter. With most of these applications, non-compliance is not going to be that much of a problem. A more likely scenario is that there will be a good amount of inconvenience.

The threat of real harm in the year 2000 situation is for large, old legacy software programs in big companies. It is going to take a major effort and cost a staggering amount to fix the software packages used by our banks, insurance companies, utility companies, and government programs such as Social Security and the IRS. If a company can't get its system up in the year 2000, it just might go out of business. The government could run even slower if they don't get their computers fixed, but this isn't really anything new. Some systems will fail and others will probably run poorly.

We will all spend more time in lines as clerks can't find our records, or the systems don't accept input. Most large businesses are hard at work fixing their hardware and software. They haven't started on their embedded systems yet. Suddenly there is a great fear that even if we fix all the software and hardware, our society will still just completely quit running when all the embedded systems fail.

Most of them will not fail. The big three auto makers are saying that their cars should run without a problem with the chips they presently have. Further testing should confirm this later in the year. But other things car related might not work. Credit systems on self service gas pumps are at risk. There is a good chance the computer diagnostic tools of non-affiliated mechanics might not work and people will have to go back to the dealer for service and pay twice as much and wait twice as long.

Where do old chips go to die? Old chips don't die. They just go on and on in lesser situations. By and large, these chips are very reliable; so reliable that the current Mars Pathfinder sending us all those neat pictures is running on a twenty year old computer chip. The venerable Z-80 chip was first sold in the 1970s. It was a wonder when it was new and you can still get them. They are much faster now, at 20 MHz, but they are still essentially the same chip doing the same work but for a much cheaper price. The Z-80 costs less than a dollar each now and the tools for programming and debugging this chip are almost universal.

In another example, the 286 chip, which still powers orphan computers in back offices, is also used in plumbing, controlling valves and flowing fluids. The 286 usually fails when it is tested for year 2000 compliance. Do you have any 286 chips about?

Some companies have production lines that are controlled by embedded processes and these firms need to get right on the problem to make sure that they can continue to produce. We need to find all the applications that use chips and make sure that they will continue to work. And this is the heart of the problem with embedded systems and also the solution. Most companies don't have to test their embedded systems or even fix them; they just have to find them. Finding them is not exactly a high tech job. You don't need to hire expensive help to solve your embedded chip problem. You need to look at your business and decide what aspects of it are mission critical and make sure those are protected first. In other words, you are going to write a disaster recovery plan for your business and while you are doing it you will solve your embedded chip problem.

Every business should have a disaster recovery plan. Almost all big businesses already have a plan and a department to oversee it. A disaster recovery plan ensures that essential business functions will continue despite most conceivable disasters. This usually entails protection of the profit and product producing parts of

the business from fire, flood, earthquake and any other problem that would negatively impact the business. All businesses should have a plan that backs up and protects the business's data, ensures continued production with minimal delay, makes sure the money keeps flowing in, and product flowing out and states the responsibilities of the company's key employees during a disaster. Sadly, few businesses have a disaster recovery plan. In fact, about the only thing in the above list that is done by most companies today is to back up their data. Most companies just hope that there will be no fire or flood that will affect them. They just hope, and, it must be said; they usually luck out. Most of the time for most companies, disasters don't happen. But the millennium change is a disaster that is going to come for everybody for sure, and this time we know exactly when it will come, down to the second. We aren't quite sure yet what it will and won't effect, but the prudent business person will begin to plan for contingencies now.

For embedded systems, this is what you do.

Start with everything that just has to keep working that either plugs in or has wires or needs batteries. This means all the systems in your buildings, your transportation, and everything that keeps product flowing out of your industry and raw materials and money flowing in. While you're at it, define every machine that is also necessary for life or health and any system that knows the date or the days of the week. Also make

sure you take a close look at systems that respond differently on the weekends such as sprinklers, traffic lights, security systems and so on. These obviously have date related software and could fail at the century change. Some industries are going to have a lot more to find than others. The health industry and hospitals have a lot of work finding and insuring compatibility with their embedded systems, which could kill people if they don't work. Companies such as construction, retail, service and supply will probably have far fewer applications to worry about. For a general rule of thumb, the more high tech you are, the longer this job will take. Ask each department what machines and systems they absolutely have to have to keep going and mark those as mission critical. When you finish this you will find that number of embedded chips out there you need to worry about has been winnowed from billions to a few dozen, maybe a few score of applications. This isn't so hard after all, and you still have more than two years.

After you have begun identifying machines and systems that probably have embedded systems, write or call the manufacturer and ask if their system or device is year 2000 compliant. They will probably not be able to answer you, or if you end up getting sales or marketing they will tell you with assurance that it is. Ask for written confirmation of that for your legal department. You need an official letter from the manufacturer stating that their system or device will work as advertised after the century changes. For most

applications the letter is all you need. Once you have the manufacturer's word, you are legally covered. If you have a system that simply cannot fail, and you need extra assurance, then you need to have some kind of back up system ready. Disaster recovery planning is very flexible and different for every business or group. If there are systems that just have to work, then spend the money to make sure they do. If you can only feel comfortable by having your devices with embedded systems tested, then go ahead and have them tested. If you think it needs to be redundant, make it redundant.

Having a disaster recovery plan in place will minimize any downtime and cut losses. Although some insurance companies are beginning to insure against year 2000 losses, don't rely on insurance. The head of a Fortune 100 Company's disaster recovery planning department told me, "Remember, insurance is the worst solution when there is a business interruption." What she meant was that insurance is usually late with payments and the money is often less than it takes to get going again, especially if there is inadequate documentation of what the business was doing before the loss. Documenting what each business unit does is also part of the plan. Insurance is necessary, but no substitute for a well thought out disaster recovery plan, which can minimize losses in profit and jobs far more than a check from the insurance company.

The embedded chip problem is really just a big warranty problem. Remember that the companies making these devices are just as anxious to have them work as you are. Product liability law has been quite clear about a company's responsibility for product failure. Years of product liability claims have demonstrated that a company has to make very sure that people are not harmed from the use, or the failure of the products they sell. And everyone knows that ignorance of the year 2000 problem is not going to be a viable excuse.

Very few firms will jeopardize their future by ignoring this issue. You need official confirmation that the makers of the chip bearing products and devices will either stand behind what they have made or will tell you how to replace them. This will take care of the issue of fixing most of these systems. Other than finding all the embedded chips in the first place, getting these letters will probably be the hardest part of the fix. For the next nine months to a year, you are going to have to explain the potential problem to a lot of people who don't have any idea what you're talking about. The person who does the asking should have enough technical knowledge about embedded systems to know they aren't getting the brush off. This will take a lot of time and will probably be very frustrating but it is the only way to get the letter of compliance. If you don't get a letter of compliance in a reasonable amount of time before the year 2000, make plans to either replace the machine, system or device, or make sure you can do without

it. Do this with all of your critical systems, then move into the non-critical areas. We will find out relatively early what to do with systems put out by major companies. By this time next year we should have a list of any vehicles that won't be compliant, and large environmental systems control manufacturers will have reports on most or all of their products. A major supplier of environmental controls just announced that they are testing all their products along with the systems their products interact with, and will complete the testing in late 1997. Then they will inform their customers and resolve any problems. They are standing behind their products. Almost every company will.

There will be hundreds of recall notices to pay attention to, perhaps even thousands. Companies will also get on the handwagon and begin advertising their products as 2000 compliant. Also by this time next year there should emerge a shared list of non-compliant or questionable products. The year 2000 problem with embedded systems is one situation where 90% compliance for non-critical systems will probably be all right, especially if there is an intelligent assessment regarding contingency plans. Programmable thermostats can often be replaced, for example, with cheaper, non-programmable thermostats. The trade off is a small penalty in efficiency.

One thing to remember with the year 2000 bug is that often, a non-technical solution is not only the easiest to accomplish, it can

also be the least expensive. There is always a tendency for technical people to think only of technical solutions when a much simpler and cheaper answer is at hand.

There is a story floating around the net about a bank with a huge vault door with an embedded chip that is impossible to reach unless the door is taken off its hinges. In order to take the door off, the whole wall of the bank must be opened up costing over a million dollars. This is probably an apocryphal story because I can't imagine anyone designing a vault door with chips you can't get at. After all, sometimes these things fail on their own. If this was the case however, the bank could still solve its vault problem for less than a hundred thousand dollars by posting an armed security guard at the vault 24 hours a day for a whole year. If it doesn't work, just leave the damn thing open. This is also a good solution for all those high dollar gated communities where the gate mechanism works with cards or codes. Most of those communities would be better places to live if they looked less like armed camps anyway. And I do not find it ironic that the computer is again forcing a more open society, this time by failing to work as planned.

If you have done what is suggested here, in a year, your critical systems will be protected. You, your job and your company should be fairly safe, at least from the embedded systems problem. The year 2000 problem big picture though, shows the dependencies we have created. It is not enough to solve our own problem in

isolation. We need to address the year 2000 compliance problems of our customers, our suppliers, and our state and local governments. If they haven't done the same thing we have, it could still be a disaster for us.

Everyone is in the same boat on the year 2000 problem. If we fix everything in our company, we are still at risk from other systems, and anyone else with whom we share data. Again the overriding message seems clear. We are all in this together and the more we help each other, the more we help ourselves.

Jon Hurrest works in Disaster Recovery Planning and is chairman of the Education Committee for the Portland Year 2000 Ready Users Group headed by Patrick Carroll. He has previous articles on the y2k in Computer Biz Magazine. <http://www.computerbiz.com>. E-mail to: jhurrest@jedi.com



Op zoek naar een uitdagende carrière bij een ambitieus, groeiend en internationaal high-tech bedrijf?

Ontdek dan Enrichment Technology

Enrichment Technology in Almelo is een hightech bedrijf dat zich bezighoudt met de ontwikkeling, productie en montage van gascentrifuges. Dit zijn complexe machines waar mee door onze klanten uranium zo kan worden verrijkt dat het geschikt is voor de opwekking van duurzame energie in kerncentrales. Daarnaast ontwerpen wij verrijkingsfabrieken die gebruik maken van onze gascentrifugetechnologie en verzorgen wij het gehele projectmanagement voor de bouw van deze fabrieken. We opereren vanuit vier vestigingen in Europa en één in de Verenigde Staten. Ons machinepark is geavanceerd, wij werken met hoogwaardige materialen en de kwaliteitseisen liggen zeer hoog.

In verband met een toenemende vraag naar nieuwe verrijkingscapaciteit op de wereldmarkt, maken we een sterke groei door. We zijn dan ook voortdurend op zoek naar nieuwe collega's die een positieve, flexibele en daadkrachtige instelling hebben en die een bijdrage willen leveren aan onze ambities.

We bieden interessante ontwikkelings- en loopbaanmogelijkheden zowel in productie, engineering als op stafafdelingen. We werken in een hightech en kennis-georiënteerde omgeving met een open en professionele cultuur.

Voor meer informatie over Enrichment Technology en carrièremogelijkheden, neem contact op met de afdeling Recruitment via werving@nl.enritec.com en kijk op onze site: www.enritec.nl.



enrichment technology
a joint venture between Urenco and Areva

www.enritec.nl



>Tim's passie voor techniek

Lees meer over Tim's passie voor techniek op

www.technolution.eu/collega

**“Mijn drijfveer
is puzzelen: het
oplossen van een
probleem op een
technisch mooie
manier.”**

Software Designer



Technolution is een projectbureau, specialist in het gecombineerd ontwikkelen van elektronica, programmeerbare logica en software voor embedded en technische informatiesystemen. In opdracht van onze klanten werken wij op ons kantoor in teams aan multidisciplinaire, technisch complexe en innovatieve (deel)systemen. Deze oplossingen zijn bedrijfs- en maatschappijkritisch en vragen om een moderne maatwerkoplossing, waarbij kwaliteit en robuustheid voorop staan.

>the right development

