

MACHAZINE

Volume 18 - Issue 2
December 2013

Wiskunde
Informatica
Studievereniging



'Christiaan
Huygens'

INTRODUCING OUR NEW COLUMNIST

Fred Vermolen

NO MORE WORRYING ABOUT YOUR LUNCH

D.I.N.S.dag

ANTONI VAN LEEUWENHOEK PROFESSOR

Eelco Visser

HELPING OUR FRIENDS AT CIVIL ENGINEERING

**Modeling on
Cement**

CONTAINING:

CURRENT AFFAIRS | ASSOCIATION | COMPUTER SCIENCE | MATHEMATICS | MISCELLANEOUS

Bedrijven- diner

***Zoek je een afstudeerop-
dracht of stageplek?***

***Kom dineren met
UL TS,
ABN AMRO en
KPN Consulting!***

16th

January 19:00

Prijs: Gratis. Schrijf je in bij het bestuur!

Locatie: De Ruif



ABN AMRO



kpn
consulting





content & colophon

MACHAZINE

is a publication of

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

Chief Editors

Hylke Visser, Cathaline Meloen

Editorial staff

Max de Groot, Friso Abcouwer, Kirsten Koolstra, Claudia Wagenaar, Rebecca Jacobs, Pieter Hameete, Anouk Rentier, Cindy Caljouw

Art Directors

Herman Banken, Hylke Visser, Floris Verborg, Cathaline Meloen

Contact address

Mekelweg 4, 2628 CD Delft

E: machazine@ch.tudelft.nl

T: 015-2782532

Concept and design

G20 Kesteren

Publisher

DeltaHage bv

Cover

<http://www.cloudsourced.com/>

Cooperating on this issue:

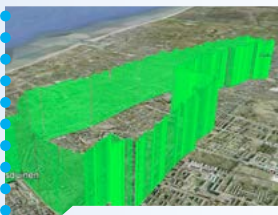
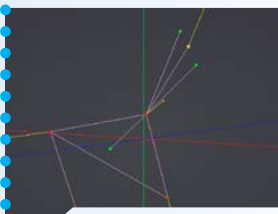
Arie Troebel, Robert Neijenhuis, Cees Witteveen, Fred Vermolen, Ginger Geneste, Jan-Willem Manenschijn, James Wafula, Beer van der Drift, Bastiaan Grisèl, Stefanie Vonk, Eelco Visser, Laurens van der Maaten, Pedro Silva, Joost Verdoorn, Mark Veraar, Domenico Lahaye, Teun Janssen, Roel Tielen, Auke Booij, Ir. Kees Zeeman

Terms and Conditions

The MaCHazine-committee and the Board are responsible for the content within this MaCHazine, in such terms that the opinion of a writer is not (necessarily) a reflection of the opinion of the committee or association. All rights reserved. No part of this publication may be reproduced, stored in a (retrieval) system or transmitted in any form or any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of the committee or association.

Advertiser index

DSW	Outside Cover
Shell	14-15
DDB	22-23
Q42	34-35



Current Affairs

Editorial	2
Agenda	2
Column: Airbag	3
Van het Bestuur	4
Column: Aftrap en Natrap	5
FSR	6
Column International Student	7

Association

TU Delft Nieuws	8
WiFi	9
D.I.N.S.dag	10
Commissie Kick-Off	12
NWERC	16

Computer Science

The Future of Programming	18
The Best of Two Worlds: Machine Learning and Visualization for Big Data	20
Graph-based Procedural Modeling	24
Dynamic Peer-to-Peer Game Networks using WebRTC	26
Java Puzzlers	28

Mathematics

Wiskundepuzzel	29
Mathematical Modeling on the Production of Special Purpose Cement	30
Noise minimalisation around on houses around airports	32
De Rogers-Ramanujan indentiteiten	36
De structuur van wilde en tamme verzamelingen	38

Miscellaneous

Gadgets - Herman Banken	40
Tracking CHanges - Bastiaan Grisèl	41
Ir. Kees Zeeman, Alumnus	42
Historical figure: Qin Jiushao	44



Editorial

Cathaline Meloen

Er waait een frisse wind door de redactie van het MaCHazine. De richting van deze wind wordt voor een deel bepaald door Hylke Visser en mij.

Dit jaar bekleden Hylke en ik de functie 'commissiehoofd' van de MaCHazine redactie. Gelukkig hoeven wij het magazine niet in ons eentje te maken; hiervoor krijgen wij hulp van een compact team. Het is geweldig om met deze groep slimme mensen ideeën uit te werken en vier keer per jaar een prachtig tijdschrift uit te brengen.

Er zullen je het komend jaar wellicht kleine veranderingen opvallen in ons MaCHazine. Naast de nodige organisatorische verbeteringen proberen wij de banden met onze schrijvers aan te halen, nieuwe schrijvers aan te trekken en artikelen per nummer beter op elkaar af te stemmen. Ook de foto's van schrijvers van vaste columns worden beetje bij beetje vernieuwd.

In dit nummer nemen we afscheid van een zeer toegewijd en gewaardeerd columnist: Cees Witteveen. Hij draagt zijn columnist-wezen met flair over aan zijn collega Fred Vermolen, via een column van beider hand, dat u kunt lezen op pagina 5. Namens de volledige redactie danken wij Cees Witteveen van harte voor zijn bijdragen aan ons tijdschrift, en heten we Fred Vermolen van harte welkom!

Verder hebben wij weer een aantal interessante bijdragen ontvangen. Onze internationale student James Wafula, die zich in ons vorige nummer onder meer verbaasde over (jawel) het efficiënte openbaar vervoer in Nederland, schrijft dit nummer weer een artikel over zijn ervaringen in Delft.

Ook vindt u een artikel geschreven door Eelco Visser. Eelco is eerder dit jaar benoemd tot Antoni van Leeuwenhoek hoogleraar, en wij zijn dan ook bijzonder verheugd dat hij voor ons magazine een artikel heeft geschreven.

Namens de redactie van het MaCHazine wensen we je veel leesplezier!

Mocht je vragen, ideeën of opmerkingen hebben, schroom niet om mij een bericht te sturen. 

Cathaline
cathalinem@ch.tudelft.nl

Activity Calendar

January

7	Procam Lecture
8	Pre-WiSpo drink
14	Alten Lecture
16	Company Dinner
21	Accenture Lecture
24	CoH Drink
31/1 - 9/2	WiFi WiSpo

February

18 - 19	De Delftse Bedrijvendagen
19	SjaarCie party
22 - 23	LAN-Party
24	General Assembly
25	Willem Jonker Lecture
26	Pré-Dies

March

3 - 7	Dies
19	Table Football Tournament



Airbag!

Arie Troebel

Lieve schatjes in het koninklijke Delft,

In het altijd drukke wetenschappelijke leven van Arie is er toch nog

altijd veel en veel werk te verstampen!

Zo was ik bezig om het werk van John von Neumann (1903-1957) even door te nemen. Deze jongen kon als zesjarige snotneus al getallen van 8 cijfers uit zijn hoofd delen. Dit lijkt voor jullie wel veel, maar ondergetekende lag op die leeftijd al een boulevard lengte voor! Maar, het moet gezegd worden, hij was werkelijk een manusje van alles, want meneer heeft veel werk verricht in allerlei categorieën.

Maar goed, ik dwaal natuurlijk weer even af. In een van de immense collegezalen van Ter-Weksel, was ik mijn schaapjes aan het onderwijzen over de grote man Qin Jiushao (1202-1261). Deze Oosterse wiskundige was ook wereldberoemd omdat hij nooit wat met wiskunde deed, en toch met wiskunde magistrale formules heeft gemaakt! Uiteraard waren al mijn studenten doodstil en totaal ondersteboven van mijn verhaal!

Na mijn college ging ik naar mijn kantoor, en wat ik niet hoopte te zien was daar wel aanwezig... Namelijk G. Flater!

"Arie, Arie, luister, ik heb een geweldig idee!", en dat zinnetje alleen al: "Ik heb een idee", dat belooft nooit iets goeds met dat brein van hem!

"Ik ga morgen de lucht in met iets heel speciaals, Arie!", zei meneer tegen mij. "En mag ik weten wat je daarvoor gaat doen, Flater?", antwoordde ik met een rustige stem. "Arie, morgen kan je het zien, en het wordt fenomenaal!"

Mijn hemel, dacht ik alleen maar...

Ik lag 's nachts te woelen in mijn bed, want als Flater iets heeft, krijg ik er weer grijze haren bij!

De volgende dag liep ik dus naar het grote veld dat achter mijn Uni ligt en wat ik toen zag! Ik zag een gigantische luchtballon, met daaronder uiteraard een bak voor een aantal personen.

"Flater, wat gaan we nu weer doen?", kon ik alleen maar met benepen stem uitbrengen. "Arie, luister, je weet dat auto's in het bezit zijn van airbags?" "Jawel", was mijn korte antwoord. "Nou, ik heb dus iets uitgevonden! Een airbag voor een luchtballon! Dus ik ga zo omhoog, als je wilt mag je mee, en eenmaal boven zet ik de vlam uit en storten we naar beneden, en wordt de klap opgevangen door de airbag die onder mijn mand is gemonteerd."

"Maar ik zie helemaal niets aan de onderkant van je mandje."

"Nee, Arie, als de ballon opgeblazen wordt dan wordt hij ook opgeblazen!"

Het moment was daar – uiteraard bleef ik op de grond staan – dat de ballon werd opgeblazen en ook aan de onderzijde van de mand vulde de 'airbag' zich. Ik pakte mijn portofoon en hield contact met Flater. Triumfantelijk stak hij beide duimen omhoog, en hij ging omhoog... "Arie, ik zit nu op 50 meter hoogte en het gas is nu op. Over."

"Flater, waarmee wordt je airbag gevuld? Over."

"Arie, dat is een detail, dat wordt ook gevuld door het gas wat nu op is. Over."

"Flater, jongen, je bent een komma vergeten met je theorie, je landing zal een beetje heel erg hard worden! Over en uit!" 🦺



Van het Bestuur

Robert Neijenhuis

Met een straffe winter voor de deur, hebben we op het moment dat ik dit schrijf nog een maand te gaan voordat het kerstvakantie is. Twee weken ertussenuit, tijd voor lekker eten, bankhangen en uitslapen.

Deze drie dingen zijn er niet echt bij deze tijd. Lekker eten is vervangen door *thuisbezorgd.nl*, bankhangen gebeurt enkel tijdens een vergadering in het commissiehok en uitslapen, dat woord ken ik al helemaal niet meer. Nouja, op de paar keer dat mijn wekker het niet doet na dan. Je kunt wel zeggen dat de eerste periode slopend was.

Maar we doen het natuurlijk voor de vereniging en dat is het meer dan waard. Wat is het toch fijn om iedere dag weer CH binnen te lopen en aan de slag te gaan met zo'n leuke groep mensen om je heen. Leden die langslipen met vragen, mensen die boeken willen kopen en gek genoeg het belletje niet kunnen vinden, of de arbodienst die langskomt om te zeggen dat die paar losse snoertjes toch echt samengebonden moeten worden. Allemaal kleine aspecten die de dagen op CH zo leuk maken: elke dag iets anders.

Als penningmeester zijn er elke dag veel dingen hetzelfde. Elke dag mag Exact weer opgestart worden, zodat de boekhouding kan beginnen. In het begin was dit toch best lastig, maar nu het eerste kwartaal erop zit denk ik toch dat ik het al aardig onder de knie heb gekregen. Ik geniet ervan om alles zo goed mogelijk te verwerken, zodat alles klopt in de boeken.

Ook bijna alle commissies zijn gestart, wat nog veel meer gezelligheid oplevert. Iedere commissie organiseert zijn eigen dingen op zijn eigen manier. Met het enthousiasme van commissies wordt je automatisch aangestoken om er ook helemaal voor te gaan. Pas geleden was de commissie kick-off waar bijna honderd commissieleden op af waren gekomen. Een groot succes! Voor sommigen is het misschien wel wat te laat geworden. Het is mooi om te zien dat zoveel mensen het naar hun zin hebben bij CH en gave activiteiten willen organiseren.

Als bestuur krijgen we ook kansen die we zeker niet aan ons voorbij laten gaan. Zo hadden we laatst de McKinsey besturendag. Meerdere besturen uit Delft waren hiervoor uitgenodigd en we stonden al erg vroeg met zes besturen op het perron van Delft om naar Amsterdam te gaan. Aangekomen in een heel hip gebouw werden we warm onthaald met allerlei lekkers. De dag stond vervolgens in het teken van het bedrijf en van ons. Uiteraard vertelden ze hoe dingen daar werken en wat er allemaal gebeurt. Daarna ging het over onze persoonlijkheden. Zo is iedereen anders; de een zegt alles wat hij denkt terwijl iemand anders wat langer nadenkt over hetgeen hij gaat zeggen. Ook besluiten nemen gaat voor iedereen anders. Sommigen beslissen vaak rationeel terwijl anderen vaak vanuit gevoel beslissingen nemen. Ondanks dat wij als bestuur zes verschillende persoonlijkheden hebben, werken we wel een jaar lang intensief samen. Samen met een oud penningmeester van CH die tegenwoordig bij McKinsey werkt, hebben we het hierover gehad. Dit was bijzonder leerzaam voor ons. De geslaagde dag eindigde in een klein Mexicaans restaurant in hartje Amsterdam.

Diversiteit zit niet alleen in ons bestuur, maar ook in de activiteiten die nog op het programma staan. Nu ik dit stuk schrijf is de CHipCie druk bezig met de laatste voorbereidingen voor het NWERC. Vorig jaar was het al een groot succes en dit jaar zijn er extra uitdagingen die we zullen moeten tackelen. Het zal wederom een groot succes worden! Maar voordat dit grote kampioenschap plaatsvindt, hebben we nog een AV en een ADSL. Tijdens de Algemene Vergadering geven onder andere de CHipCie en de Symposium commissie een presentatie over hun voortgang. Door de ADSL wordt de afstand tussen docent en student kleiner, wat natuurlijk geniaal is. Iedere dinsdag is er nog steeds een lunch activiteit. Dit is onze nieuwe aanwinst waar we heel tevreden over zijn.

Naast de net genoemde activiteiten staan ons nog heel veel meer dingen te wachten de komende tijd. Wat mij betreft komen we die straffe winter wel door. Mocht je er toch moeite mee hebben, kom dan gezellig een kopje koffie of thee op CH drinken. Tot dan! 🍷



Aftrap en Natrap

Cees Witteveen en Fred Vermolen

In dit artikel stelt de oude columnist, Cees Witteveen, de nieuwe columnist, Fred Vermolen, voor.

Het leek me een goed idee om mijn afscheid van deze kolom eens op een bijzondere manier aan te kondigen. Al een paar maanden be kroop me het gevoel dat het mooi is geweest. Na alle fratsen die ik zonder externe review in het MaCHazine geplaatst kon krijgen wilde ik dit plezier ook eens aan een ander gunnen. De keuze voor een opvolger was niet moeilijk. In zeker opzicht deelt mijn opvolger hetzelfde gevoel voor licht absurde humor als ik mij zelf toedicht en bovendien was hij voor een aantal jaren een verdiepingsgenoot. 's Morgens lieten wij al vroeg op de 7e etage ons licht schijnen over allerlei prangende kwesties en hadden we na een kwartiertje praten overal een oplossing voor. Ik weet bijna zeker dat zijn toenmalige oplossingen in toekomstige columns een tweede leven gaan leiden. Ook qua omgangsvormen hebben we denk ik voor een novum op de 7e etage gezorgd. Zo hebben we een overdreven Japanse stijl van buigen voor elkaar geïntroduceerd, waarbij Fred regelmatig zijn kleren vuil heeft gemaakt door languit op de vloer te gaan liggen. We vinden het beiden jammer dat deze beleefdheidsvorm niet doorgevoerd is in de rest van de faculteit.

Beste lezers, het is tijd geworden om jullie nieuwe columnist aan jullie voor te stellen.

Wie ben jij nu eigenlijk, Fred?

Het zijn meestal anderen die iets over je kunnen of durven te zeggen. Daarom beperk ik me zo veel mogelijk tot het feitelijke. Ik studeerde Technische Natuurkunde aan de TU-Delft en ben daarna gepromoveerd in het modelleren van bewegend randwaardeproblemen met toepassingen in de metaalindustrie. Tijdens dit promotiewerk werkte ik een half jaar aan een Max Planck Instituut in Duitsland. Daarna ben ik via het CWI in Amsterdam in de groep Numerieke Wiskunde aan de TU-Delft beland. Mijn interesse ligt in modelleren, eindige elementenschema's en analyse van (partiële) differentiaalvergelijkingen en in de stochastische differentiaalvergelijkingen. Mijn toepassingsgebieden zijn in de medische biologie, poreuze mediastromingen, en metaalkunde. Daarin lopen een aantal projecten waarin ik promovendi begeleid. Verder geef ik met veel plezier twee vakken in de wiskundeopleiding. Ik heb mijn draai hier gevonden omdat de combinatie van onderwijs en onderzoek mij zeer bevalt. Het is een voorrecht om met enthousiaste en creatieve geesten te werken en het beste uit deze mensen te halen.

Waarom wil je eigenlijk columns schrijven?

Op de eerste plaats ben ik door jou benaderd met de vraag of ik je op wilde volgen als 'columnist' voor het MaCHazine. Het lijkt me leuk om dit te doen omdat ik dan op een lichtvoetige manier iets van mijn gedachten op papier kan zetten. Ik hoop dat de lezer er stof tot nadenken van krijgt en dat er zo nu en dan een glimlach op zijn of haar gezicht verschijnt. Ik steek ook graag de draak met (onnodige, onzinnige) formaliteiten. Misschien zal een column ook een manier kunnen zijn om mijn hart eens te luchten over zaken waar ik me druk over maak. Wie weet heeft deze column dan zelfs wel een therapeutische werking voor mij.

Waar maak je je druk om?

De vraag 'Waar maak ik me niet druk om?' zou wel eens meer op zijn plaats kunnen zijn. Ik maak me het meest druk over onrecht en zinloze geldverslindende projecten. Dit kunnen grote, eindeloze aanbestedingen en projecten zijn van de overheid, maar deze kunnen zich net zo goed ook dichtbij op de universiteit en in het onderwijs manifesteren. Een bekend voorbeeld is dat krijtborden in collegezalen vervangen werden door smartboards die niet (goed) werken. Ik neig ernaar sceptisch te staan tegen door hogerhand opgelegde veranderingen, maatregelen of wijzigingen. Verder maak ik me zorgen om de kwaliteit van het onderwijs op de middelbare school en op universiteiten. Dit lijkt een Nederlands probleem maar ook internationaal ziet men in veel landen gelijksoortige tendenzen (met uitzondering van Finland). Ik denk dat de neergang nog niet gestopt is, en dat studeren in de toekomst weer enkel voor de rijkere klassen haalbaar wordt. Naast dat dit een persoonlijk drama kan zijn, zal Nederland mogelijk veel talent en dus de spreekwoordelijke boot missen. Talentvolle en gemotiveerde mensen zouden de gelegenheid moeten hebben om te studeren ongeacht de financiële staat van hun ouders. Het zou me niet verbazen als kinderen uit volgende generaties gemiddeld minder hoog opgeleid zullen zijn dan hun ouders. Als we hiervoor veel ambachtlieden voor terug krijgen, hoeft dit niet eens zo erg te zijn, maar het is nog maar de vraag of dit laatste dan ook zal gebeuren. De nadruk zou meer op kwaliteit dan op kwantiteit moeten liggen. Innovatie moet naast een middel ook een doel zijn, en dat bereik je door slimme, eigenwijze en creatieve mensen met ballen. Om even na te trappen op de column uit [1]: "Als Nederland te lui en te dom wordt, dan zijn we makkelijk te foppen en wordt het vaak dokken".

Fred, na zulke fraaie woorden geef ik graag het stokje aan jou over. Doe je best. 

Referenties

[1] Cees Witteveen, Column, In: Machazine, volume 18, issue 1, 2013.

FSR

Ginger Geneste & Jan-Willem Manenschijn

De eerste tentamenweken zijn voorbij en elke student verlangt alweer naar de Kerstvakantie. Er zijn dit collegejaar een hoop veranderingen doorgevoerd en sommige veranderingen moeten nog komen, dus ook de FSR heeft niet stil gestaan afgelopen kwartaal!

Laten we voor de zekerheid eerst nog kort toelichten wat de FSR nou eigenlijk is. De Faculaire Studenten Raad (FSR) is een medezeggenschapsorgaan binnen onze mooie faculteit EWI. Bij beslissingen die binnen de faculteit worden genomen vertegenwoordigt de FSR het belang van studenten. De FSR werkt samen met het faculteitsbestuur en de onderwijsdirectie om de kwaliteit van het onderwijs en de faciliteiten op de faculteit te verbeteren. Klachten en suggesties horen we dan ook graag: deze kun je mailen naar fsr@ch.tudelft.nl.

In het laatste bericht van de FSR dat in het MaCHazine stond hebben we ons beleidsplan voor komend jaar gepresenteerd. In het kort omschreef het plan dat er dit jaar de nadruk gelegd word op de volgende drie onderwerpen: De kwaliteit van het onderwijs, de algemene studentenzaken en faciliteiten op de faculteit. In dit artikel zullen we een aantal subonderwerpen uitlichten waar we nu druk mee bezig zijn. Deze onderwerpen hebben we ook besproken met het faculteitsbestuur tijdens de laatste overlegvergadering. Bij deze vergaderingen is onder andere de decaan aanwezig, die altijd graag luistert naar onze input en dit ook zeer waardeert!

Afgelopen kwartaal is de kwaliteit van het onderwijs nauwlettend in de gaten gehouden, met name het verloop van de nieuwe curricula en het verdwijnen van de witte week. Over het algemeen leken de studenten voldoende op de hoogte te zijn van de veranderingen en is het afgelopen kwartaal in principe goed verlopen. Toch zijn er een aantal punten opgemerkt die eventueel verbeterd kunnen worden. Deze punten gaan we goed uiteenzetten, voorbereiden en vervolgens bespreken tijdens de volgende overlegvergadering met de decaan. Mocht je willen weten wat deze punten precies inhouden, er over mee willen denken of je input willen geven, dan kun je ons altijd aanspreken in de wandelgangen of op CH, waar Jan-Willem Manenschijn meestal zit.

Voor de algemene studentenzaken heeft de FSR recent een overleg gehad met de studieadviseurs. Tijdens dit gesprek is er onder andere gesproken over de bereikbaarheid van de studieadviseurs en hoe we ervoor kunnen zorgen dat de bereikbaarheid goed genoeg blijft met oog op de toenemende studenten-aantallen. Ook zal er in de toekomst een hoop veranderd worden rondom de faciliteiten van onze faculteit. Zo is er een begin gemaakt aan de geplande verbouwing. Deze verbouwing heeft bij sommige studenten geluidsoverlast veroorzaakt. Facilitair Management en Vastgoed heeft ons verteld dat het geen probleem is om de verbouwers er op aan te spreken als er sprake is van geluidsoverlast tijdens een college. Zij zullen het geluid op dat moment tot een minimum beperken, of zelfs stoppen met de werkzaamheden. Dus mocht je overlast ervaren: meld dit dan even bij de werkenden en dan zullen ze ermee stoppen!

Tevens is de FSR druk bezig met het zogenaamde 3S systeem. Studiekeuze, Studiekeuzecheck en Studiestart. Vanuit Den Haag zijn er nieuwe regels opgesteld: de zesdeklasser moet zich voor 1 mei hebben ingeschreven. Daarna



heeft de aankomende student recht op een studiekeuzecheck, waarbij ze kunnen checken of ze de goede studiekeuze hebben gemaakt. De TU Delft heeft daar weer regels voor gemaakt, waar wij ons als faculteit aan moeten houden. Dit houdt in dat we alle aankomende studenten een digitale vragenlijst laten invullen. Hier kan de faculteit dan een persoonlijk advies uit proberen te halen. Dit kan een positief, twijfel of negatief advies zijn. Om een dergelijke vragenlijst tot stand te laten komen, heeft de faculteit input van studenten nodig. Hier kan de FSR inspringen. Wij zullen met de faculteit meedenken over de inhoud van de matching. Deze moet zo opgesteld zijn dat we een duidelijk beeld van de verwachtingen van de student krijgen. Ook zal de motivatie en de kennis van de student getest moeten worden.

Als de aankomende student een twijfel of negatief advies heeft gekregen, zou de student bijvoorbeeld kunnen worden uitgenodigd voor een dag op de faculteit. Hier zal de student op een concrete manier kennis maken met de studie en de faculteit. De invulling van deze dag is tot op heden onbekend, en ook hierbij zal de FSR meedenken over de invulling.

Zoals je waarschijnlijk al hebt kunnen bedenken, zitten er vrij veel haken en ogen aan dit project. De FSR zal er op toezien dat de pilot zo goed mogelijk verloopt en geen negatieve gevolgen heeft voor de huidige of aankomende studenten.

Tijdens het komende kwartaal zal de FSR de bovenstaande punten verder uitwerken en eventueel nieuwe punten op zich nemen. Ben je benieuwd naar de wekelijkse voortgang binnen de FSR? Like dan onze Facebook-pagina "FSR EWI". Heb je vragen, opmerkingen of input, kun je altijd mailen naar fsr@ch.tudelft.nl of langslopen op CH en vragen naar Jan-Willem Manenschijn!



Column International Student

James Wafula

Time is flying, quite literally. Somebody wrote that stress and old age seems to make us feel like time is speeding up. I hope he was wrong. The semester is soon coming to an end and as expected, there is more to be done. It is one of those challenging times for many students, but at the same time quite interesting because this is precisely the reason we are here.

A few years back, I overheard somebody asking what drives a person to even think of doing Mathematics. I started thinking about it and asked myself what would be the appropriate answer to this question. I have come to learn that Mathematics is about structure, it is a universal part of humanity, because humanity thrives on imagination, and imagination is structured.

The power of imagination

I grew up in a sleepy village where back then, before modern technology came calling, elderly people did not know how old they were. If you asked an elderly person how old she was, she would tell you which planting/ harvesting season she was born, and then tell you how many seasons had passed, or tell you a major event that happened around the time of their birth, and tell you how many cycles of that event had occurred. This system was the only one they knew, and was the order of the day for that generation in my sleepy village. In some way, everyone was expected to know this. Precision in determining their age was expected since many activities in their lives revolved around age groups. This goes to show just how structure has always been part of us. These people had a way of mapping what they knew onto their brains, and using the power of imagination to solve day to day challenges.

Discovering the future

Today, we live in a world of supercomputers. These systems are driven by algorithms, a huge part of which is mathematical in nature. We have come a long way, but still a lot has to be done. Many planets are yet to be discovered. Just to try and quench the unquenchable human thirst. Kepler is still out there and more mathematics still needs to be done to make use of her invaluable power.

It can be argued that computer systems can take up most of procedural tasks. But, Mathematics continues to thrive because human beings are still imagining! The theory of everything is yet to materialize, or let us just say that more mathematics is yet to be done.

Have a lovely autumn. 🍂

TU Delft Nieuws

Max de Groot

De TU Delft is de grootste en oudste Nederlandse publieke technische universiteit, opgericht door Koning Willem II op 8 januari 1842. Maar wat gebeurt er momenteel nu verder op de TU Delft? In dit artikel zetten we de belangrijkste gebeurtenissen van de afgelopen tijd op een rijtje.

Nuna wint eindelijk weer

Het Delftse studententeam werd onder grote belangstelling van familie, vrienden, relaties en media feestelijk onthaald op Schiphol na het winnen van de World Solar Challenge in Australië op 10 oktober 2013. Het Nuon Solar Team deed voor de zevende keer mee aan het wereldkampioenschap zonnereizen en won de wereldbeker voor de vijfde keer. Het team van Tokai University won de afgelopen twee edities. Voor het eerst sinds 2007 is Nederland nu weer wereldkampioen.

Zelfrijdende auto's

Veilig op een korte afstand van je voorganger rijden waarbij de auto zelf de snelheid bepaalt en waarbij de bestuurder geen handen aan het stuur heeft. Automatische auto's zijn volop in ontwikkeling, en nu is het tijd om de techniek en de interactie met gebruikers te evalueren in complexe omstandigheden op de publieke weg. Daartoe hebben TU Delft, RDW, Connexxion en TNO het Dutch Automated Vehicle Initiative (DAVI) gestart. De officiële lancering van DAVI vond plaats op 12 november 2013 tijdens de Innovatie-estafette in de RAI Amsterdam. Minister Schultz van Haegen stapte direct na de opening van de beurs zelf in een DAVI-voertuig en reed mee over de A10.

Uniek aan deze demonstratie is dat het hier gaat om automatische coöperatief bestuurd auto's. Dit betekent dat de voertuigen niet alleen beschikken over techniek om automatisch te rijden, maar ook uitgerust zijn met technieken om te communiceren met andere voertuigen en de infrastructuur.

Algae Food & Fuel investeert in Delfts algen-onderzoek

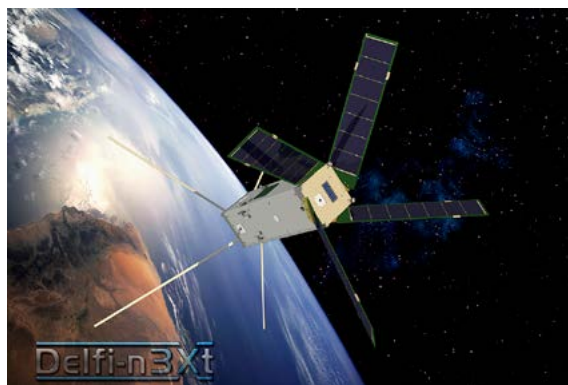
Het bedrijf Algae Food & Fuel zal de komende zes jaar een belangrijke investering doen in algen-technologie die door de TU Delft is ontwikkeld en gepatenteerd. Algae Food & Fuel wil de Delftse technologie opschalen en op termijn een productielijn voor algenolie opzetten. Van die algenolie is onder meer biodiesel te maken. Het contract tussen Algae Food & Fuel en de TU Delft wordt bezegeld tijdens de Innovatie-estafette, op dinsdag 12 november in de RAI te Amsterdam.

Algen zijn interessante kandidaten voor de grootschalige productie van olie. Onderzoekers van de TU Delft zijn intensief met dit onderwerp bezig en hebben onlangs een slimme methode ontwikkeld waarmee ze uit alle algen de dikste, en dus meest geschikte, soort kunnen vinden. Het uiteindelijke doel is om hiermee olieproducerende algen zo dik mogelijk te maken, ze vervolgens uit te persen, er biodiesel van te maken en daar auto's op te laten rijden.

Universiteiten gaan meer leraren opleiden

En dat wordt hoog tijd, vinden ze zelf. "We hebben het onderwijs enigszins verwaarloosd", zegt voorzitter Karl Dittrich van universiteitenvereniging VSNU. Net als het kabinet en de Tweede Kamer willen ook de universiteiten dat er meer academici leraar worden. Academisch geschoolde leraren zijn niet alleen goed voor het basis- en voortgezet onderwijs, meent Dittrich, maar ook voor de universiteiten zelf.

Volgens critici verwaarlozen universiteiten vooral hun eigen onderwijs ten gunste van wetenschappelijk onderzoek. Moet daar dan ook iets aan gebeuren? "Ik zie wel een kentering. Er is een roep om de productiedruk te verminderen en weer meer aandacht voor onderwijs te krijgen. Ik snap die beweging. Het onderzoek en onderwijs aan de universiteit moeten met elkaar verweven zijn en we moeten de balans zoeken."



Figuur 1: De Delfi-n3Xt

TU Delft lanceert satelliet Delfi-n3Xt

Op 21 november om 08.10 uur, is Delfi-n3Xt gelanceerd vanuit Yasny in Rusland. Om 09.47 uur wist 'Mission Control' in Delft voor het eerst succesvol contact te maken met de satelliet. Bijna vijf jaar hebben studenten en medewerkers van de TU Delft gebouwd aan Delfi-n3Xt, de tweede Nederlandse universitaire satelliet. Een Russische Dnepr-raket bracht Delfi-n3Xt samen met tientallen andere satellieten naar de ruimte. De satelliet is ontwikkeld in samenwerking met de Nederlandse ruimtevaartindustrie. Het ruimtevaartbedrijf ISIS, een succesvolle Delftse spin-off, verzorgde de lancering vanuit Rusland. Delfi-n3Xt is de opvolger van de eerste TU Delft satelliet Delfi-C3, die in april 2008 werd gelanceerd. Hoewel ontworpen voor een jaar, functioneert de satelliet nog steeds. De Delfi-satellieten zijn slechts 34 cm hoog en 10 cm breed, niet veel groter dan een melkpak. Ondanks de kleine afmetingen zijn de functionaliteiten van de satellieten vergelijkbaar met die van hun grote broers.

Referenties

- [1] <http://www.tudelft.nl/nl/actueel/laatste-nieuws/>
- [2] <http://www.delta.tudelft.nl/nieuws>
- [3] <https://intranet.tudelft.nl/directe-links/nieuwsberichten/tu-nieuws/>



WiFi

Beer van der Drift

Ooh die herinneringen! Partijen in het zwembad, adten in de glijbaan, gek doen in het bubbelbad. Ooh, en de busreis naar Frankrijk! Jäger op de piste, koken met je huisje, uitgaan in jodeloutfit, leren boarden. En dit mag ik samen met de rest van de WiFi organiseren! Whiiiooe!

Hai! Ik ben Beer, voorzitter van de WiFi '13/'14. Waar het voor staat? Wintersport, Feest en Iets!! Vorig jaar werd de WiFi opgericht na de succesvolle lustrum wintersport en dit jaar mogen wij, met z'n 6en en onze sexy QQ'er, de organisatie van dit evenement en nog meer op onze schouders nemen. Wij zijn al vanaf het begin van het collegejaar ontzettend druk bezig met bellen, overleggen etc etc. Dit allemaal om de meest geweldige activiteiten van CH neer te zetten!

Het begint allemaal met het zwembadfeest. Vorig jaar gingen we met een grote groep internen en externen ontzettend hard 'zwemmen' in CHarlois in Rotterdam met als thema "Zuipen of Verzuipen".

Uiteraard na een biertje en een avondmaal in de /Pub gingen we met z'n allen samen naar een zwembad vol opblaasbeesten, bier en studenten, en het was net zo awesome als het klinkt. Dit jaar gaan we een nog vetter feestje neerzetten op 13 december, jaja, vrijdag de 13e! Een ongeluk zit in een klein hoekje, pas maar op met je zwembroekje. Op dit moment zijn we druk druk druk bezig met de voorbereidingen van het feest, maar op het moment van lezen is het feestje al geweest. Ik ben benieuwd!

Ons hoofdevenement is natuurlijk de wintersport! Vorig jaar kreeg ik als sjaarsch maar om m'n oren dat ik mee moest gaan met CH dat jaar, maarja ik kende de vereniging nog niet super en had niet zoveel geld... Maarja he, wintersport had ik altijd wel eens willen doen en leren, want ik was, naast een keer met paps en mams op m'n 6e, nooit op wintersport geweest. Een paar mensen bij elkaar geraapt, huisje gestart (ik als huis HJ...) en gaan! Na al een veel te enthousiaste nachtelijke busreis en geslapen te hebben in het gangpad, kon het allemaal pas echt beginnen. Overdag boarden, dan après-skieën, eten met het huisje of anderen en dan weer terug voor het avondfeest!

Er zijn 2 dingen waardoor ik één van de geweldigste weken van afgelopen jaar heb gehad. Als eerste het boarden zelf, ik had het nooit gedaan, maar na een week les met een awesome groepje CH'ers (YOLO!) gingen we al met volle snelheid de rode piste af. De kick van het boarden, de omgeving, de parties midden op de pistes, te gek voor woorden.

Het andere wat me is bij gebleven, is de groep zelf. Ik heb het gevoel dat ik CH pas echt heb leren kennen in die wintersportweek. Je bent dan niet alleen samen op het hok en op feestjes, maar echt een week lang met z'n allen. Die week heb ik zo ontzettend veel mensen, mede sjaarsch en ouderejaars, leren kennen die ik anders nooit had leren kennen. Dit draagt nu nog steeds bij de rest van m'n leven met CH. De gezelligheid en sfeer is echt geweldig! Het voelde alsof ik met een vriendengroep van 50 man op vakantie was. De random borrelavond bij de andere huisjes, jäger op het balkon, wat mooi.

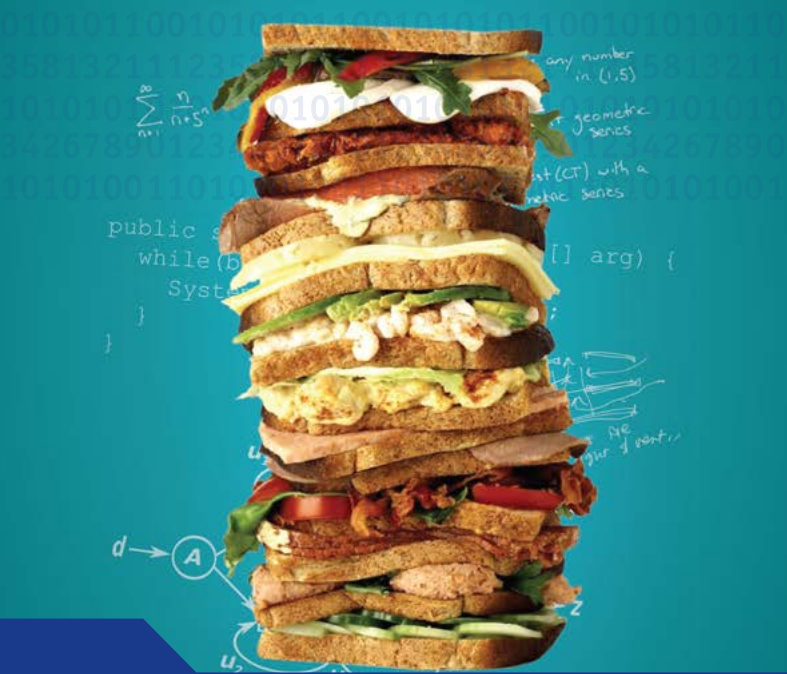
Goed, genoeg over de herinneringen, dit jaar is er weer een wintersport en gaan we naar Risoul, ook in Frankrijk. Wil je mee? Schrijf je dan heel snel nog in, 20 december sluit de inschrijving en als ik het goed heb is dat de dag van het uitbrengen van deze MaCHazine! Voor de mensen die zich hebben ingeschreven, super dat jullie meegaan, je zal geen spijt krijgen! We zijn al druk bezig om een te vette wispo neer te zetten inclusief alle vette dingen eromheen.

Onze laatste activiteit is het 'iets' gedeelte. Vorig jaar was dit een (ijskoud) strandfeest met barbecue! Dit kwam jammer genoeg door de lange winter, maar de kou hield sommige helden niet tegen om toch het water in te springen! Lekker met heel CH weer de terugreis gezelligmaken met bierspelletjes in de tram zonder bier haha! Wat wij dit jaar gaan organiseren dat weten we nog niet. Dat is nog te ver weg. Wel kan ik je vertellen dat we echt iets geweldigs gaan neerzetten!

De datum van dit 'iets' is nog uiterst illuster en dat zullen we later bekend maken. Wintersport is direct na het 2e kwartaal dus dat duurt nog wel even, nu eerst het zwembadfeest! We gaan met de WiFi zorgen dat dit een topjaar wordt, we gaan er echt wat geweldigs van maken! 🏊‍♂️



Association



D.I.N.S.dag

Bastiaan Grisèl

De Dinsdag Is Niet Smeren dag. Iedereen die de afgelopen maanden op de verenigingsruimte langskwam werd bij wijze van spreken doodgegooid met dit nieuwe concept. We hebben inmiddels het eerste kwartaal achter de rug, en de eerste lezingen zijn georganiseerd. In dit artikel blikken we terug op de eerste paar lezingen en kijken we hoe de introductie van dit nieuwe concept bevalt.

Voor wie de afgelopen tijd onder een steen heeft geleefd, of voor de niet meer zo actieve leden van CH, nog even een korte uitleg van de Dinsdag Is Niet Smeren dag (D.I.N.S.dag). Omdat we vorig jaar hebben gezien dat er vooral vanuit bedrijven enorm veel interesse was om lezingen te geven en we merkten dat we enorm gave lezingen moesten afzeggen omdat we simpelweg geen plek hadden in de jaarplanning, hebben we dit jaar besloten om bijna elke dinsdag in de lunchpauze een lezing te organiseren. Dit zijn zowel lezingen van bedrijven, als ook lezingen van hoogleraren en professoren. Om de lezingen zo toegankelijk mogelijk te houden, hebben we ervoor gekozen om gratis lunch te verzorgen voor degenen die zich van tevoren inschrijven. Inschrijven kan zowel via een inschrijffijl op CH als op internet. Omdat dit erop neer zou komen dat CH elke dinsdag de lunch verzorgt, leek het een logische stap om ook de ledenlunches op dinsdag te organiseren. Omdat een ledenlunch niet op financiële compensatie uit het bedrijfsleven kan rekenen, blijft de prijs hiervoor gewoon 1 euro. In de tentamenperiodes worden er geen lezingen georganiseerd, we blijven ten slotte een studievereniging.

Voor de eerste lezing van Bol.com hadden we zaal Chip gereserveerd, een zaal waar iets meer dan 100 man in kon. We konden onze ogen niet geloven toen we de inschrijffijlsten aan het overtypen waren en het totaal op 184 man kwam te staan; een legendarisch hoge opkomst voor een lezing van CH en een logistiek drama voor het serveren van de lunch. Hoe moesten we bijhouden wie wel en wie geen lunch had, en hoe gingen we die lunch zo efficiënt mogelijk verdelen? We hebben uiteindelijk gekozen voor een oplossing waarbij iedereen

een reserveringscode toegestuurd kreeg per email en deze werd dan aan de deur gecontroleerd. Dat controleren moest natuurlijk snel gebeuren en met meer dan 180 man zou een papieren inschrijffijl niet toereikend zijn. Er is daarom een hip appje geklust waarmee we via twee tablets de reserveringscodes konden invullen en checken. Dit bleek soepel te werken maar vooral het pakken van de broodjes na binnenkomst bracht de nodige overhead met zich mee. We hadden namelijk platters (een hip Engels woord voor een schaal met broodjes) gekozen met verschillende soorten broodjes, met als gevolg dat iedereen –ondanks aanmoedigende woorden van het Bestuur– eerst rustig elk broodje bekeek, om er vervolgens twee te pakken, toch weer eentje terug te leggen en weer een andere uit te kiezen. Uiteindelijk zat iedereen rond 12:50 in de zaal en kon de lezing beginnen.

Bol.com

De lezing van Bol.com ging over hoe ze in hun webshop de instant search hebben geïmplementeerd, zodat de site begrijpt dat je graag GTA5 wil kopen ook al heb je alleen nog maar de g in de zoekbalk getypt. Niels vertelde ons over de eerste versies van het algoritme, waarbij de relevantie van een bepaald artikel bepaald werd door hoe vaak mensen ernaar zochten. Schijnbaar wilden klanten op dat moment graag een Nijntje Dekbedovertrek, waardoor deze vaak gezocht werd en daardoor ook hoger in de zoekresultaten van andere klanten terecht kwam. Het vervelende was alleen dat het Nijntje Dekbedovertrek helemaal niet meer verkocht werd door Bol.com en dat vele klanten en bestelling plaatsten voor een product dat niet leverbaar was. In een volgende versie van het algoritme werd ook meegenomen hoe vaak mensen daadwerkelijk het product kochten, in plaats van alleen hoe vaak erop gezocht werd.

Om alle data die voor dit algoritme nodig was op te slaan en te verwerken, is er gebruik gemaakt van Apache Hadoop (hadoop.apache.org). Hadoop is een stuk software dat er voor zorgt dat berekeningen kunnen worden uitgevoerd op grote datasets door deze te distribueren over meerdere computers. Omdat de berekening gebruik maakt van de kracht van meerdere computers, kan je in plaats van dure supercomputers meerdere goedkope computers kopen. Zo kreeg Niels voor een interne test van zijn systeem een aantal afgedankte computers ter beschikking. Eén dergelijke computer zou nooit zo'n complexe berekening aan kunnen, maar zet er een aantal naast elkaar en Hadoop doet het werk. Zelfs toen tijdens de demo een voeding doorbrandde waardoor er één computer onbruikbaar werd, vingen de andere computers dit op en als gevolg duurde de berekening weliswaar iets langer, maar was het resultaat nog steeds correct. Door simpele relaties aan te brengen in de honderden gigabytes aan userdata die verzameld werd was het mogelijk om met een relatief simpel MapReduce algoritme relevante zoekwoorden te voorspellen.



Een overvolle zaal bij de eerste lunchlezing door Niels Basjes van Bol.com



Deze eerste D.I.N.S.dag-lezing wist een zaal van meer dan 180 man te boeien en bood zowel interessante kijk in de keuken van de grootste webwinkel van Nederland maar focuste vooral op de toepassing van een nieuwe technologie op het gebied van big data processing.

McKinsey

De week na het inhoudelijke praatje van Niels Basjes deed McKinsey uit de doeken hoe het leven als consultant in elkaar steekt. Een luchtig praatje waarin alle vragen die mensen hadden over het salaris, de werkuren en de inhoud van het werk aan bod kwamen. Het was jammer dat er weinig inhoudelijks over projecten verteld kon worden en het interactieve element van de lezing kwam met meer dan 100 man in de zaal niet helemaal goed uit de verf. Voornamelijk voor de mensen die nog niet eerder met een consultant in aanraking waren gekomen was het praatje een leuke introductie.

Bayesiaanse Netwerken in de Rechtzaal

Ronald Meester kwam in de volgende D.I.N.S.dag lezing vertellen over hoe redeneringen in de rechtzaal op een correcte manier kunnen worden gevalideerd. Hij legde uit dat een netwerk in kaart kan brengen wat van elkaar afhangt en welke onzekerheid met die afhankelijkheid gepaard gaat. Deze zogenaamde Bayesiaanse netwerken vinden vele toepassingen, en leiden soms tot zeer tegen-intuïtieve uitkomsten. Stel bijvoorbeeld een jury bepaalt dat een verdachte niet schuldig is aan een zeker zedenmisdrijf. Na deze uitspraak wordt de jury geconfronteerd met het feit dat de verdachte eerder is veroordeeld voor eenzelfde soort misdrijf. Een intuïtieve reactie is, dat deze nieuwe informatie de kans groter maakt dat de jury met de vrijspraak een fout heeft gemaakt. In veel gevallen is een intuïtieve reactie onjuist, en de kans dat de jury een correcte uitspraak heeft gedaan, wordt in



veel gevallen juist groter door de nieuwe informatie. Al met al een interessante lezing die ook laat zien hoe een fundamenteel wiskundig onderwerp in de praktijk toegepast kan worden.

Q42

Na het succes van de vorige lezing van Q42 over het gebruik van Google App Engine in hun eigen projecten, werd ook in deze lezing een hip stukje nieuwe techniek uitgelegd: Meteor (www.meteor.com). Meteor is een web framework waarbij op een eenvoudige manier interactieve web-applicaties kunnen worden gemaakt. Rahul legde de zaal uit hoe hij de site van Q42 omgebouwd had naar Meteor en bouwde vervolgens zelf een klein demo-appje. Zelfs in deze kleine demo-app kwam de kracht van Meteor al naar voren. Veranderde Rahul iets aan de code, dan werden alle open browsers automatisch geüpdatet door de Hot Code Push in Meteor. Zo waren er nog heel veel andere gave technieken die voor informatici direct toepasbaar waren in verschillende projecten en voor wiskundigen een interessante kijk in keuken bood van een echte app-developer.

UL Transaction Security

UL TS is voor veel CH-ers een onbekende naam en het was dan ook onduidelijk wat men precies moest verwachten van deze lezing. Snel werd al duidelijk dat ze bij UL enorm veel gave projecten hebben lopen, met name op het gebied van mobiel betalen. Hugo vertelde hoe we vroeger met onze pinpas door de terminals heen swipeten en via onze magneetstrip data overbrachten en betaalden. Door de opkomst van fraude en skimming is de EMV chip geïntroduceerd en we gingen van het swipen naar het dippen (doelend op het feit dat je je pinpas tegenwoordig in de pinterminal moet steken). Hierdoor heeft het geen zin meer om de magneetstrip te kopiëren, alleen de EMV chip wordt door geldautomaten gebruikt. UL kwam voornamelijk vertellen over een nieuwe technologie waarbij ook het dippen tot het verleden behoort en we er tappen voor terug krijgen: contactloos betalen. Een app op je telefoon moet de bankpassen gaan vervangen. Hugo vertelde dat de technologie inmiddels veilig is om in de praktijk gebruikt te kunnen worden en in Leiden is er dan ook al een proef uitgerold voor 1000 early-adopters. Hij vulde aan dat het op dit moment het lastigste is om de volwassen eindgebruiker ervan te overtuigen dat deze technologie niet alleen "leuk is voor de kinderen" is maar ook de toekomst van het mobiel betalen.

ASML

"An introduction to microchips and analysis of complex systems" was de titel van dit praatje over de technieken die ten grondslag liggen aan de grote apparaten die ASML bouwt. Er werd uitgelegd hoe een chip grofweg gemaakt wordt en dat een machine van ASML in essentie een grote en enorm nauwkeurige dia-projector is. De precisie die vereist is voor een goed werkende chip is ongeveer 10 nanometer, dat is ongeveer hoeveel een grasspriet groeit in een seconde! Wederom een enorm interessante lezing over waar ze zich op de ASML campus mee bezig houden en wat de uitdagingen zijn waar ASML tegenaan loopt.

Een goed begin

Al met al kunnen we terugblikken op een succesvol kwartaal met veel leuke en inhoudelijke lezingen. De bezoekersaantallen zijn inmiddels gestabiliseerd op een man of 80, dus wat eerst een logistiek drama was is nu ineens een geoliede machine geworden. Komend kwartaal staan er weer enorm leuke lezingen op het programma en met de opgedane kennis en positieve reacties kan het niet anders dat het elke dinsdag weer enorm vet gaat worden. 

Association



Commissie Kick-Off

Stefanie Vonk

Twee bestuurders op de bank, een onder zijn tafel en de rest deed alsof

ze nog wat konden. Het was een ware uitputtingsslag, maar wat was de

commissie kick-off gezellig! Veel commissies waren voltallig aanwezig en

anderen deden alsof ze een hele commissie waren.

Het begon allemaal lang geleden achter een bureau vol belangrijke papieren. Hier werd lang gepeinsd en afgewogen wie er dit jaar alle ontzettend gave commissies moesten gaan vullen. De lijsten werden doorgelopen en kloppend gemaakt, en waren uiteindelijk compleet. De mails werden verstuurd en de commissies werden definitief vastgelegd.

Toen was het natuurlijk tijd om met al deze mensen het jaar goed te beginnen. De bioscoopzaal werd gereserveerd en het menu werd samengesteld. De QQ'ers nodigden al hun commissies uit en de kaartverkoop ging van start. Na een aantal spamsessies in alle vergaderingen schreven uiteindelijk meer mensen dan ooit zich ruim van tevoren in. Natuurlijk waren er nog een paar mensen die op het allerlaatste moment nog op het lijstje wilden, en daar werd uiteraard nog plek voor gemaakt. De officiële uitnodiging stond op de mail en iedereen wist wanneer hij waar moest zijn, oftewel: de Kick Off was een feit.

Dinsdag 12 november 17:34. Daar gingen ruim 80 commissieleden op de fiets naar de Pathé. Er ontstond meteen een file en de verkeersregelaars konden het bijna niet aan. Eindelijk aangekomen bij de bioscoop begon het commissiegevoel al goed te leven. Er werden gezichten geschminkt en hoeden opgezet. Na de aanschaf van de nodige bakken popcorn en grote hoeveelheden beugels werden de eerste studenten al de zaal ingelaten en de film werd gestart. Feuten, het feestje. En een feestje was het. Een grandioze weergave van een intens overdreven visie op het studentenleven werd aan ons vertoond, wat natuurlijk zeer vermakelijk was. Nog wat tips naar aanleiding van de film:

mocht je in de /Pub een drankje willen bestellen, bestel dan vooral 2 cola en geen bier. Dit kan natuurlijk alleen als je al naar binnen mocht, want toegang tot de /Pub is natuurlijk ook enkel voor Leden.

Na de film werd de fiets weer gepakt en werd Google Maps opgestart, want waar zat die stadsherberg ook alweer? Toen de locatie dan eindelijk door iedereen gevonden was, konden we naar binnen en werden de tafels gevuld. Voor velen was het een schok toen er wel borden op tafel werden gezet, maar geen bijbehorend bestek en zij tot de conclusie kwamen dat ze toch echt met hun handen moesten eten. Het breekbrood dat aan het begin van de maaltijd op tafel lag, viel bij iedereen in de smaak. Er werd luidruchtig gegeten, in een verbazend tempo gedronken en weer bijgeschonken, gelachen en geluisterd naar de mooie verhalen die verteld werden.

Op het menu stonden naast heel veel vlees ook selderie en spekkies, hoewel de spekkies natuurlijk alleen voor de belangrijkste commissies van CH, want wat zouden wij als studievereniging toch moeten zonder onze collegeresponsiegroepen (Bravo!)? Toen alle glazen weer goed gevuld waren, werd er nog een voordracht gehouden door onze voorzitter over hoe vet het is om een commissie te doen, wat iedereen natuurlijk alleen maar kon beamen.

De avond leek op z'n eind te lopen en de cadeaus werden met plezier ontvangen. Maar iets in mij zei dat de avond nog lang zou duren en dat klopte helemaal, want het was tijd voor de afterparty.

Er werd lang gediscussieerd, maar de meeste

stemmen gingen toch naar het Proeflokaal. Een (net niet)oude lullentafel, een WiFitafel, een AkCietafel en nog een heleboel andere tafels die ik vergeten ben, zorgden er voor dat CH goed vertegenwoordigd aanwezig was.

Na enige navraag ben ik tot de conclusie gekomen dat de volgende speciaalbiertjes zeker aan te raden zijn: Weihestephaner Vitus, de Straffe Hendrik en de Gulden Draek. Mocht je hier nog vragen over hebben, kan ik je altijd doorverwijzen naar onze bierkenner Robert. Toen ook het Proeflokaal besloot dat het tijd was dat iedereen zijn bedje moest opzoeken, klonk dat als een zeer goed idee in mijn oren.

Echter, toen ik vanochtend op CH kwam bleek het dat niet iedereen diezelfde gedachte had gehad de avond ervoor. Onder andere een bestuursgenoot, een aantal sjaars en een paar verdwaalde studenten hadden besloten een afterparty te houden en dat was te zien. Mocht je je afvragen waarom die studenten allemaal zo moe uit hun ogen keken, dat kwam dus daardoor, want ik hoorde dat er is gefeest tot de zon weer opkwam. Ik kijk nu al uit naar de commissiebedankdag, die vast net zo gaaf gaat worden! 







Working in IT for an Energy Company

Shell

What does it look like to work for a company that does not have software as its focus?

When you are studying Computer Science, people often believe you will end up programming once you graduate. Most people realize later that this isn't true, but what does your job look like then? There are many companies that are not creating software solutions but do have significant challenges in the IT area. One example is Royal Dutch Shell, one of the biggest companies in the world, where data is measured in petabytes for some applications, and working in IT means finding the best IT solution for whichever business you are supporting: from marketing to huge engineering projects. In this article, three new employees discuss what their role is at Shell IT.

Ulas Ulgen *IT Consultant*

As our technology gets better, it constitutes an increasingly bigger proportion of business. As a result thereof many technical and business people find themselves in interface roles between those two domains. These hybrid roles are not appealing to everyone, and it does require a generalist mindset. Although a lot of professionals are pushed into these positions because of business needs, some people enjoy it. I am certainly one of those people. In fact, even before knowing that there was a need for these kinds of people, I already shaped my education accordingly. When I first arrived in Delft in 2005, I started my studies in TBM, following an IT Domain. I set my eyes on management of technology. Later in life (by talking to many people), I realized that managing something that I don't fundamentally understand was not something that I wanted to do. That's why I changed my Major to Computer Science but reserved my Minor for Project Management both in TU Delft.

My work as an IT Consultant is good mix of IT and business skills prepared me for a life of doing IT without IT. More specifically, I am working in a team of very experienced IT consultants who drive IT standards on big engineering projects that you see on National Geographic. Although I am not programming, I am benefitted immensely by my study. For example, when working on an

integrated planning for the new design tool, it really helps if you know how time consuming it is to setup a decent user acceptance test. It really helps to know the pitfalls for designing an Expert system with waterfall method. It really helps to be able tell a web-developer half way across the world that you are an engineer yourself and that you both know that two-week activity he scheduled should really take shorter.

Being computer savvy does not only mean "Not being fooled". When I joined Shell, our team did not yet have a website. When the website was delivered to us, it did not fulfill all our requirements. I contacted the development team they told me that I had to submit a chance request. This included filling out a 4 page document, asking for a budget, when (if) approved waiting for resources to be assigned to our project, requirement elicitation and implementation. The whole process was to cost a lot of money, and cost us around 3 months. Instead I "looked under the hood", adapted the code here and there, and viola! Within 2 weeks our website was good to go.

Anne van Ee *Application Delivery Analyst*

In 2012, I graduated from Computer Science in Delft and started my job at Shell. During my studies, some of my favorite courses were about human computer interaction. I ended up doing my thesis in the data visualization area which was also very interesting. To my surprise, the human computer interaction courses have been the most useful in my current role. I say surprise because during my studies, the focus was not on these courses at all.

I currently work for our Loyalty business (most will know this as Air Miles here, but I support any Shell loyalty program we have in Europe). I make sure our team is aligned with other companies about sending out marketing communication to our customers. This is where the human computer interaction comes in because the business always wants to change something! My role is to help determine the requirements for the change and align between the business and the developers to find a suitable solution. What I think one of the



coolest things in my role is having the freedom to see opportunities to improve processes and to be able to drive the changes for such improvements even if you only have been working in the team for a few months!

We have a database with our customers and transactions and have secure file transfers with the other companies to make it possible to send out an email or paper mail to our customers. This is not too difficult, but the complexity is in the size, the constraints and the exceptions (not code exceptions, but business rule exceptions!). We support about 12 countries and each has their own market specifics with regards to our marketing campaigns: different loyalty programs, different sizes of the markets and different site (retail/fuel station) systems within every country. Size is important because queries can run for hours and we avoid emailing more than 1 million customers per day to reduce the load on our website.

As an IT professional, you try to standardize your solution as much as possible while still trying to keep it flexible for market specifics. This is the real challenge because an easy change in, for example, Germany may not be as easy to implement in Slovakia. Once the change is done in Germany, you can often bet they will request something similar in other countries when the business colleagues hear about it. It is, therefore, important to be prepared to deploy the solution in other countries when designing solutions in a specific market because you want to keep the landscape of your IT solution aligned with the other countries.

Radoš Lazarević *Technical Application Specialist SMART*

For my bachelors, I studied Computer Engineering at the Faculty of Electrical Engineering, University of Belgrade. During studies, as all my friends were preparing to start a career as Software Developers, I wasn't completely comfor-



table. I wanted to know more about the everyday application of what we were being taught to do, which had little to do with everyday uses of computers and more to do with theory and high academia. As much as I loved the art of programming for the feeling of accomplishment when you solve a complex logical problem or a code issue, I had a feeling that if I become "just" a developer I will be looking at only one piece of the puzzle, rather than the whole puzzle itself. This, and a strong passion for exploring foreign countries, is what led me to enroll in the IMMIT (International Master in Management of IT), an Erasmus Mundus master program that was a cooperation of three Universities: University of Marseille (IAE-Aix), University of Turku (Turku School of Economics), and Tilburg University. Apart from being a wild ride in the intercultural sense, this master program equipped me with exactly what I was missing – business application of computers. Unlike my bachelor's studies where we learned about designing Operating Systems, Software Compilers, and grid systems for Distributed Processing, here we learned about Macroeconomic, Off-shoring, Lean, etc.

Currently, I support the software applications that are used by engineers in Shell. I belong to an IT organization in Shell called TaCIT (Technical and Competitive IT), which deals with IT solutions that are made exclusively for Shell (and in many cases by Shell too). These IT solutions are Shell proprietary and bring Shell the competitive advantage on the market. My team is called the "3rd line support", which means that there are two layers of support before us, so the issues that trickle down to us are usually complex, and in many cases impossible to solve – i.e. bugs. This is why we are placed within a larger team that also consists of Developers (people who make software) and Product Owners (people who provide the input on what exactly to make and who ultimately own it). I chose this role as a good starting-point to get to know the vast world of Shell IT, since the environment I'm currently working in is very familiar from my bachelor studies. Later I plan to slowly "move away from the code" into roles that have a more to do with Project Management, and a strategic view on where the IT in Shell is going to. I am fortunate that in the small planet that Shell is, all these roles are available to me, as well as first-class mentors who can help guide me in my career.

One of the coolest things that I tried quite recently is called "Telepresence". It is (an extreme version of) video conferencing technology that allows me to connect with colleagues in a lifelike way. There are special rooms with several huge LCD screens, and when you start the meeting, your colleagues will appear there in natural size and in HD resolution, virtually as if you are sitting in the same room. No lag, no breaks, and the resolution so high you can see every detail on their face and every facial expression. I remember that as a kid I was watching Sci-Fi movies and cartoons, and I was wondering if I will ever get to use something like this.

Summary

Working for a large company like Shell means there is a lot of IT to support many different parts of the business. Possibilities range from supporting engineering projects you see on National Geographic, or working with applications created to help engineers at oil fields, to supporting marketing people when sending out newsletters to loyalty customers. Not only can you support different businesses within one company such as Shell, but you can also take on a variety of different roles as an IT professional. You can take on a 3rd line support role like Radoš and help solve complex issues when they arise in an application, or you can sit between the business and the IT organization to ensure what is developed matches the needs of the business like Ulas does. The nice part is that you do not even need to know ahead of time where your passion lies because throughout your carrier, you can choose roles that cover a broad range of businesses and IT roles.



NWERC

Rebecca Jacobs

The weekend of the 22nd November, the NWERC was held at the TU Delft for the second year in a row. NWERC stands for North West European Regional Contest and it is a programming contest. The contest is meant for students from all over North West Europe and it is a preliminary contest for the World Finals, where students from all over the world will compete to become the world's best programmer.

The preparations for the NWERC had started several months ago. Since then the CHipCie, the organizing committee, has been working to prepare for the contest. But on Friday the 22nd of November the NWERC 2013 was about to begin. After a full day of working to completely rearrange the large rooms of building 35 at the Cornelis Drebbelweg number 5, the contestants started to arrive in small groups. Since the contestants come from countries as far away as Iceland, some of them had had quite a long journey. So after a cup of coffee and subscribing, they could go to their hotel or go and explore Delft a little bit. During the subscriptions, the committee was still working on the contest arrangement. Since this could be seen from the hallway, quite a few of the contestants and coaches already tried to get a glimpse of the table arrangement. After all the contestants subscribing on the Friday evening had received their T-shirt, badge and goody bag it was time to put the finishing touches on the contest room and then it was time to go home for the committee and everyone helping out.

The Saturday morning meant an early start for everybody helping out. Coffee had to be made, cookies had to be put out for the contestants and the late registration had to be opened. After everybody had arrived again at building 35, they had to be lead to EWI for the opening ceremony. During the opening ceremony speeches were given by several people. Among them was Thomas Verwoerd, the regional contest director and Peter Kluit, a jury member and a honorary member of W.I.S.V. 'Christiaan Huygens'. After all the contestants and their coaches were warmly welcomed to Delft and to the NWERC 2013 and all other announcements were made, they could go back to building 35 where the test session would start. The test session is a small competition to test the system. All teams and coaches are allowed to inspect the workspaces and the system. There were also some assignments for them to solve so that they could practice turning in their solutions to the jury.

After the test session everybody enjoyed lunch, existing out of sandwiches, which were handmade on-site by everybody helping out at the NWERC. After this the contestants and coaches could participate in excursions which were organized by the committee, or they could go and explore Delft on their own some more. The activities organized by the committee were a beer tasting at 't Proeflokaal in the city center and a guided tour of the city centre. Some contestants even went to The Hague for the afternoon.

At six everybody had to be back, because the program continued. At lecture hall A in EWI a lecture was given by a speaker from TomTom. He had an interesting talk on what TomTom does, and what things they do better than

the competition. He explained how navigation systems have developed the last several years, what they possibly could achieve in the near future and how modern technology constantly creates more functionality for navigation systems. After the lecture the jury and technical staff made some last announcements on things that had come up during the test session. During the lecture the food had arrived at building 35. So when the last remarks were done, all contestants and coaches could walk to building 35 to enjoy a fine Chinese diner while enjoying a beer or a glass of wine. Most contestants left early to get a good night's sleep for the contest the next day.

The contest on Sunday started at half past nine in the morning. The contestants participate in teams of 3 persons. Each team got one laptop to program on. Furthermore they received paper and pens to program on paper. The goal of the competition is to solve as many problems as possible. The problems are provided by the international jury. The teams can print their code so they can look at it on paper. Also, for each correct solution they receive a helium filled balloon. The team which solves the most problems wins, or when there is a draw, the team which it took the least time to solve the problems wins. When a team hands in an incorrect solution to a problem, they receive a time penalty, but only when they solve the problem later on. If the problem is not solved by that team during the contest there is no time penalty.

The competition takes five hours and during the competition lunch was served to the contestants. The committee had bought kruidnoten as a special Dutch snack. Also during the competition small gatherings for coaches where held so they could comment on the contest and advise on how the contest should be organized next year, when NWERC will be held in Linchköping, Sweden. After four hours of contest the scoreboard is frozen and no balloons are handed out anymore. This means that at the prize ceremony, the winner is not known yet. After the contest all teams and their coaches could get a drink from the bar while the jury explained how the problems could have been solved. After that a tense award ceremony started. Both the first and the second teams are guaranteed a place at the World Finals in Yekaterinburg next summer. The second place went to a team from the University of Cambridge named Beuler, and the first place went to a team that was playing a home game, it was a team from the TU Delft called geen.opdracht5. The team members were: Jan Elfers, Robert van Staveren and Harmjan Treep. On behalf of the complete organization of NWERC 2013: congratulations, boys!

I would like to thank everybody who helped during the weekend, and I would like to express a special thanks to board 57 of W.I.S.V 'Christiaan Huygens' for helping out before and during the contest. 





The Future of Programming

Eelco Visser

Software systems are the engines of modern information society. Our ability to cope with the increasing complexity of software systems is limited by the programming languages we use to build them. Bridging the gap between domain concepts and the implementation of these concepts in a programming language is one of the core challenges of software engineering. Modern programming languages have considerably reduced this gap, but often still require low-level programmatic encodings of domain concepts. Or as Alan Perlis formulated it in one of his famous epigrams: "A programming language is low level when its programs require attention to the irrelevant". The current generation of general purpose programming languages such as Java were designed to build a wide range of applications for the Von Neumann architecture. These languages are not adequate for the emerging new computing landscape.

New Hardware

First the computing hardware landscape has radically changed from the traditional Von Neumann machines with a processing unit with a local memory hierarchy. New multi-core computers provide multiple processing units so that many computations can be executed simultaneously. Using multi-core machines effectively requires redesigning programs to be concurrent, steering away from the race conditions and dead locks that are easily introduced in such programs. Furthermore, computers do not operate in isolation, but are connected through local and global networks into distributed systems. To make use of this architecture, programs need to be split up into multiple parts that communicate with each other over the network. This requires deciding how to split programs and adds the overhead of designing and implementing communication protocols. Some, and increasingly more, of these machines in the distributed system are mobile devices, which are not reliably connected to

the network and have limited battery power that needs to be used sparingly. Maintaining overlapping state on different nodes in the network, requires reasoning about the (eventual) consistency of data.

New Applications

This hardware substrate has given rise to new applications and new software architectures to support them. For example, web applications are prime examples of distributed systems with parts running in the browser, the server, and the database. Each of the tiers in these systems is programmed in different languages (HTML, CSS, and JavaScript in the browser; Java, Ruby, or PHP on the server; and SQL on the database), while the programming environment that web programmers use is not aware of their composition. Another emerging class of software is data analytics programs to analyze the enormous amounts of data produced by, for example, scientific experiments, social media, and network traffic. For specific application domains this often results in accidental complexity, i.e. complexity due to the encoding of domain concepts in the language, rather than intrinsic complexity of the problem at hand. The standard abstraction mechanisms provided by programming languages are not adequate; many libraries are languages in disguise, lacking proper checking of safety properties.

New Programmers

The combination of new types of applications running on new types of computers makes software development more challenging than ever before. In addition, many new programmers are not trained as such. There is a wide range of programmers with different backgrounds, ranging from systems programmers, application programmers, data scientists, and end user programmers (using spreadsheets) without formal training as software engineer.

New Programming Models and Languages

Programming these machines requires encoding instead of expressing a solution for a domain problem. Changing the configuration of the hardware platform may require radically changing the implementation details of the program. The challenge for the software engineering and programming languages field is to design new programming models and programming languages that allow us to abstract over the complexity the underlying machinery, separating the functionality of applications from the details of their implementation. For example, reactive programming provides a solution to the distribution and scalability problem by making programs event-driven.



Domain-specific software languages (DSLs) address the complexity problem through linguistic abstraction by providing notation, analysis, verification, and optimization that are specialized to an application domain and allow direct expression of domain concepts. DSLs provide high-level abstractions specialized to a domain of computation, which allows developers to directly express design intent ('language shapes thought'), and which allows a compiler to report errors using domain terminology. Linguistic integration of DSLs for different concerns enables finding cross-concern safety violations and prevent late detection of failures. For example, our WebDSL web programming language integrates sub-languages for persistent data models, user interface templates, access control policies, and data validation. The IDE for WebDSL gives immediate feedback about cross-concern safety violations such as incorrect references to the data model in user interface templates.

Software Language Design and Engineering at TU Delft

The mission of the Software Language Design and Engineering research program at TU Delft is to enable software engineers to effectively design, implement, and apply domain-specific software languages (DSLs) for emerging domains of computation. However, a software language is a complex software system in its own right, consisting of syntactic and semantic analyzers, a translator or interpreter, and an interactive development environment (IDE), and can take significant effort to design and implement.

Language Workbenches are a new breed of language development tools that considerably lower the threshold for software engineers to develop DSLs. A language workbench is an integrated development environment (IDE) to support the design and implementation of programming languages. By means of high-level meta-languages, a language engineer defines the syntax and various semantic aspects of a language. The workbench then generates an implementation of a compiler, interpreter, and/or IDE for the language.

Examples of modern language workbenches include EMFText, MPS, and Xtext. My group at TU Delft has developed the Spoofox Language Workbench, based on research contributions in compositional syntax definition, syntax error recovery, program transformation, code generation by model transformation, concrete object syntax, and test-driven language development. Despite great progress in the last decade, the community is still investigating the best abstractions for defining languages.

In Delft we are investigating DSL engineering from three perspectives. In the language design track we investigate approaches to the systematic design of DSLs with an optimal tradeoff between expressivity, completeness, portability, coverage, and maintainability. To understand the design process we perform DSL design case studies. For example, in the domain of web programming we have developed the WebDSL language. WebDSL linguistically integrates the various concerns of web programming, which in other approaches are covered by separate, loosely coupled languages. This is not just a prototype effort. We have used WebDSL for example to develop the WebLab programming education web application that is used in several courses at TU Delft and TU Darmstadt.

In the language engineering track, we investigate the automatic derivation of efficient, scalable, incremental compilers and effective IDEs from high-level, declarative language definitions. The goal is to minimize the effort for the language designer and to maximize the quality of the generated implementation. Recently we developed a new declarative approach to the definition of the name binding rules for a language, i.e. the rules that determine which uses of identifiers belong to which declarations. Based on such rules we can now generate an incremental name resolution algorithm, a task that previously required a considerable programming effort.

Supported by a prestigious grant from the NWO VICI program, we have recently started a new research track on semantics engineering. In this track, we investigate techniques for verification of properties of language designs. For example, we would like to automatically prove (or show a counter example) of the type soundness of a language. That is, if a program is approved by the type checker for a language it should not exhibit run-time type errors. Thus, a large class of programming errors can be reliably captured at compile-time.

Symposium

On the occasion of my appointment as Antoni van Leeuwenhoek Professor, TU Delft will host a symposium on the future of programming on Thursday January 16 and Friday January 17 2014. The symposium will provide an overview of challenges in software development and programming languages and visions to their solution from different angles. The symposium features a lineup of renowned national and international speakers from academia and industry including Markus Püschel (ETH Zürich), Markus Völter (Itemis), Erik Meijer (TU Delft, Applied Duality), Sebastian Erdweg (TU Darmstadt), Arie van Deursen (TU Delft), Stefan Hanenberg (Universität Duisburg-Essen), Manuel Serrano (INRIA), and Brandon Hill (Oracle Labs). The symposium is concluded by the inaugural speech of Eelco Visser. Students interested in the developments in programming languages are most welcome to attend the symposium and inaugural speech. 

For more information see:

[1]: <http://eelcovisser.org/wiki/future-of-programming>

The Best of Two Worlds:

Machine Learning and Visualization for Big Data

Laurens van der Maaten

Data science and Big Data are hot topics in computer science today. Data science develops algorithms that allow companies to extract knowledge or value from the massive data sets that companies gather.

Such massive data sets are everywhere: just think about all the data that Facebook has gathered about you and your friends, or about the data Albert Heijn gathers via its (new) Bonuscard. Both these companies presumably know when was the last time you gave a party at your student house: Facebook because on the day after the party many of your friends posted party pics (and perhaps tagged you in them), and Albert Heijn because you bought beer the day before the party and aspirin the day after. The knowledge you can extract from massive data sets is very valuable, and the type of knowledge you may obtain is often much more unexpected than in the above example.

A nice example of this is the American supermarket Target, which is gathering information similar to Albert Heijn and is using this information to send its customers targeted marketing. A father of a teenage daughter complained at a Target just outside Minneapolis, because Target kept sending his daughter coupons for maternity clothing, which he found very inappropriate. A few days later, the father sent Target his apologies: he had just found out that his daughter was actually pregnant. How did Target know the girl was pregnant before her father? Apparently, she recently bought cocoa-butter lotion and a purse large enough to fit diapers. Previous Target customers who bought these products also bought maternity clothing, which is why Target sent the coupons. Target could even predict the girl's due date with a small margin.

There are many other examples of what you can do with data science: Google knows about a flu epidemic spreading before doctors do, because people do flu-related queries before going to the doctor; Netflix knows which movies you like after you provided just a few movie ratings because it always has a few customers that gave very similar ratings as you; and Merck can predict the effectiveness of new medicines because it knows the results of earlier medicine trials. Such knowledge can be turned into value: Google can sell flu predictions to pharmaceutical companies so they know where to ship their medicines to; Netflix can increase the chance that you rent a movie; and Merck can speed up the development of new HIV medicines.

How do you analyze Big Data?

You are probably convinced by now that data science offers great possibilities. So how do you actually do data science? First, you need lots of data that are stored in some database (or on a Hadoop cluster if the data is truly big). Second, you need ways of analyzing this data. Roughly speaking, there are two possible ways in which you can analyze data: (1) you can use *visualization* algorithms and see whether you can spot some interesting patterns — visual analytics; or (2) you can implement *machine-learning* algorithms that automatically recognize patterns in your data — predictive analytics. Both approaches have advantages and disadvantages. Visualization can give you insight into your data, but it is impossible to visualize more than just a few variables at a time. This is a problem when your data comprises 20,000 gene expressions (variables) from various tissue samples: no visualization can show which types of tissue exhibit similar expressions across all genes. Machine

learning is designed to recognize patterns in data with lots of variables, but the models and predictions it generates are usually incomprehensible.

My research aims to obtain the best of both worlds by combining machine learning with visualization: I am interested in developing machine-learning models that, when they are used on a large and complex data sets, facilitate interpretable visualizations of the results of this analysis. In particular, my research focuses on an approach called *embedding*. In embedding, one represents each (high-dimensional) object by a point in a two- or three-dimensional space. These points are arranged in such a way that similar objects are represented by nearby points, whilst dissimilar objects are represented by distant points. The locations of the points are learned by moving around the points in the space in such a way as to minimize the discrepancy between similarities between the objects and similarities between the points. The main advantage of the (automatic) embedding analysis is that the results of the analysis can readily be visualized: for instance, the learned point locations can be used to draw a scatter plot (a kind of map) that shows which of the original input objects are similar and which ones are dissimilar.

t-SNE

The main question in embedding is how to measure the discrepancy between the similarities of the N original objects and the similarities of the points in the map. With Geoffrey Hinton, I have developed such a discrepancy measure, which has led to a technique called t-SNE [1]. Suppose we are given the similarity between two objects i and j , called p_{ij} , and without loss of generality, suppose that these similarities are normalized to sum up to one¹. As a result, we can interpret the similarities p_{ij} as a probability distribution over all pairs of objects (i, j) . The probability of picking a particular pair of objects depends on the similarity of these objects: more similar objects are more likely to be picked under the distribution. The best way to obtain the values for p_{ij} depends on the problem: if the input objects are high-dimensional vectors, one may compute a “normalized Gaussian kernel”; if the objects are events, the value of p_{ij} may be the probability that event i and j co-occur; etc.

Now assume that we want to represent each object i by a two-dimensional map point $\mathbf{x}_i \in \mathbb{R}^2$. The computer needs to determine the locations of all the N map points $\mathbf{x}_1, \dots, \mathbf{x}_N$ in such a way that the distances between them appropriately represent the “input” similarities p_{ij} . To do so, we define a second set of similarities that measure the similarity between point i and j :

$$q_{ij} = \frac{1}{Z} \frac{1}{1 + \|\mathbf{x}_i - \mathbf{x}_j\|^2}. \quad (1)$$

Herein, the variable Z is normalization constant that makes sure that the similarities q_{ij} are probabilities (i.e. sum up to one) just like the p_{ij} 's. If you paid attention in your statistics class, you may recognize the above expression as a Student-t distribution with one degree of freedom. If you didn't pay much attention, you may still notice that q_{ij} will be large whenever the squared Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|^2$ between them is small, and nearly zero whenever their pairwise distance is large. So the map similarities q_{ij} correspond to the input similarities p_{ij} whenever high similarities between the original objects are modeled by small distances in the map.

¹We also assume that each similarity p_{ij} is non-negative: $\forall i, j : p_{ij} \geq 0$.

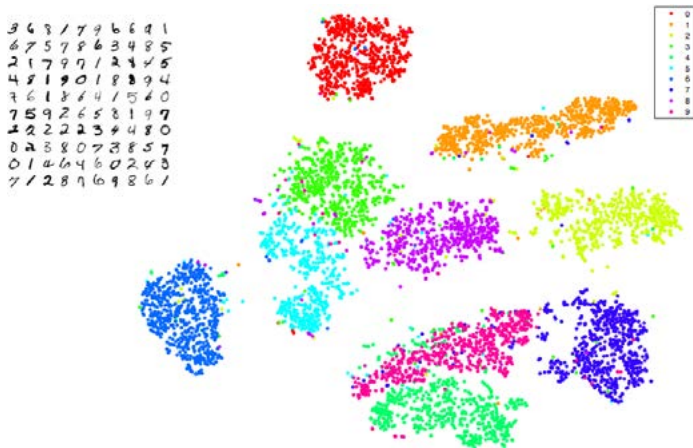


Figure 1: Map of handwritten digit images (see inset) constructed by t-SNE.



Figure 2: Map of exactly the same data constructed by PCA.

Only one question now remains.... where do the map points $\mathbf{x}_i$ come from? Recall we want to set the map points such that large p_{ij} 's are modeled by large q_{ij} 's. Hence, we can measure the quality of any two-dimensional map by comparing the q_{ij} 's of that map with the p_{ij} 's we received as input. The measure we use in t-SNE is a function called the *Kullback-Leibler divergence*. It is the natural distance measure between two probability distributions:

$$C(\mathbf{x}_1, \dots, \mathbf{x}_N) = \sum_{i=1}^N \sum_{j=1}^N p_{ij} \log \frac{p_{ij}}{q_{ij}}. \quad (2)$$

It is not too hard to confirm that this function will be relatively large when similar objects (large p_{ij}) are modeled by distant map points (small q_{ij}). Therefore, good maps are maps for which the value of $C(\cdot)$ is small. We can use standard optimization techniques (gradient descent) to find such a map.

What can you do with t-SNE?

What can you do with t-SNE? Simply put, you can take a big data set of complex (high-dimensional) objects and build a map of these objects. An example of such a map is shown in Figure 1. The input objects here are images of handwritten digits; each image has 784 pixels. The similarity p_{ij} is defined based on the number of overlapping pixels of the two images i and j . These similarities are used to construct a two-dimensional t-SNE map that can be plotted as a scatter plot, as shown in Figure 1. In the figure, each point corresponds to a digit image, and the color of the point corresponds to the digit shown in that image. The t-SNE plot clearly shows that digit images form separate clusters, as may be expected if you think about digit images in terms of pixels: zeros look very different from ones. However, when this result was published, it was unprecedented. As a reference, Figure 2 shows a map

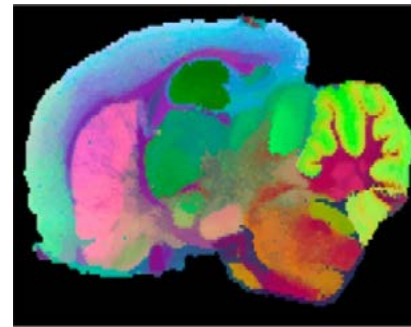


Figure 3: Map of a rat brain constructed by applying t-SNE on gene expression data. Figure adopted from [3].

constructed by a competing technique, called principal components analysis (PCA), on exactly the same data. You may decide for yourself which map you think provides a more accurate representation of the underlying digit images. Similarly to the digit example, Albert Heijn can use t-SNE to make a map of all their customers; perhaps to find out that students form a separate cluster with a distinctly different shopping pattern. Or Target could use a t-SNE map to show the angry father that his daughter is truly in a cluster of pregnant women (and find out what pregnant women tend to buy).

It is important to realize that the maps that t-SNE creates need not necessarily be visualized as scatter plots. For instance, one may define the map points $\mathbf{x}_i$ to be three-dimensional, and use the resulting map points as an RGB color code. Figure 3 shows an example of such a visualization (the visualization was made by Judy Fonville *et al.* [3]). The underlying data is gene expression data measured at every pixel location in a slice of a rat brain. Since there are about 20,000 genes, we thus have a measurement vector with 20,000 variables at every location. The locations are treated independently to create a three-dimensional t-SNE map in which each point corresponds to a location in the rat brain. The resulting map locations are then treated as RGB color codes to create the rat brain map. The figure gives great insight into which regions of the brain have nearly the same gene-expression profiles (and therefore, presumably have the same function) and which brain regions have very different gene expressions (and presumably, a different function).

Because it makes such nice pictures, t-SNE is now widely adopted both in industry and in academia: it is used for data analysis by major companies including Google, Yahoo!, Microsoft, and Siemens, as well as by a range of scientific studies (for instance, in brain research, cancer research, computer security, and bioinformatics). Rumor has it that even the American Department of Homeland Security is using t-SNE to visualize their terrorist databases.

Want to know more?

For more information, you can read [1] or watch the TechTalk I gave at Google [2]. Do you want to get involved in data science yourself? Courses related to data science include: Big Data Processing and Datamining (at the B.Sc. level); and Pattern Recognition, Data Visualization, and Machine Learning (at the M.Sc. level). If you want to write your thesis on a data-science topic, please feel free to shoot me an email with your ideas!

References

- [1] L.J.P. van der Maaten and G.E. Hinton. Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research* 9(Nov):2579-2605, 2008.
- [2] <http://www.youtube.com/watch?v=RJVL80Gg31A>
- [3] J. Fonville, C.L. Carter, L. Pizarro, R.T. Steven, A.D. Palmer, R.L. Griffiths, P.F. Lalor, J.C. Lindon, J.K. Nicholson, E. Holmes, and J. Bunch. Hyperspectral Visualization of Mass Spectrometry Imaging Data. *Anal. Chem.* 85(3), 2013.



Kick start your career!

Every year 'De Delftse Bedrijvendagen' leads 2200 students to the start of their career. If you want to hit the ground running this year, make sure you do not miss this event. Whether you are job hunting or in need of an internship, 'De Delftse Bedrijvendagen' offers an unique opportunity to get to know a wide variety of high-profile companies. 'De Delftse Bedrijvendagen' is the largest technology oriented career event in the

Netherlands, and takes place each year in the Aula Conference Centre of the TU Delft. This career event is specifically aimed at the students of Delft University of Technology and offers them a wide range of companies and institutions. 'De Delftse Bedrijvendagen' kick starts your career with the Application Trainings, Presentations Days, In-house Days and the Interview Days.



Prepare yourself for your job interview

On the **11th, 12th and 13th of February** 'De Delftse Bedrijvendagen' offers Application Trainings. The companies will provide you with both general tips and tricks as well as detailed personal advice.



Get to know over 100 national and international companies

On the **18th and 19th of February** 'De Delftse Bedrijvendagen' hosts its most well-known event, the Presentation Days in the Aula Congress Centre. This fair gives you the opportunity to get to know more than one hundred and forty different national and international companies! There are several ways of meeting these companies. You can visit their information stand or view their company presentation. There is also the opportunity to have your résumé checked. Each day ends with informal drinks together with the recruiters.

Take a look behind the scenes

From the **17th of March until the 4th of April** the In-house Days offer an opportunity to form a better, more complete idea of the companies you are interested in. Your résumé will be forwarded to the companies of your choice. Based on the résumés the companies will select the participants. As the name suggests, the In-House Days are a great way to learn on location about the culture of a company and the projects they work on. On **March the 20th** an In-House Day will be held in the Aula Congress Centre in Delft for companies situated far away from Delft. You can sign up for the Inhouse days until the **25th of February** - the Tuesday after the Presentation Days.

Take part in one-to-one interviews

From the **23rd of April until the 9th of May** 'De Delftse Bedrijvendagen' will be completed with the Interview Days. During this period, companies are free to invite students for an interview or simply to get to know the company in more depth. For the Interview Days your résumé will be sent to companies you are interested in, but also to companies that are interested in your degree specifically. You will receive an overview of which companies are interested, and from these you can make a selection.

design: Studio Piraat

Participation

All activities described above are included in a single price when participating in 'De Delftse Bedrijvendagen'. On top of this, you also receive a full-color company guide with information about all participating companies. You can participate by signing up via our website, www.ddb.tudelft.nl, or by coming to the Aula Congress Centre on the **21st, 22nd or 23rd of January**.

Participation is only € 10,- if you register before **January 23rd**. From **January 24th** and onwards participation will cost € 15,-. All personal information will be treated strictly confidential.

So if you are looking for a job, an internship or a graduation project, from January on you can subscribe at www.ddb.tudelft.nl!

Organization

'De Delftse Bedrijvendagen' is organized by five study societies, that together form 'The Pentagon':

- Vereniging voor Technische Physica
- Gezelschap Leeghwater
- Technologisch Gezelschap
- VSV 'Leonardo da Vinci'
- W.I.S.V. 'Christiaan Huygens'



Graph-based Procedural Modeling

Pedro Amorim Brandão da Silva

In this article I will present the current result of my ongoing PhD thesis project, part of which has been developed at the Computer Graphics and Visualization Group. By manipulating interactive graphs, my work allows designers to define procedural 3D modeling guidelines for automatic generation of large virtual cities.

The creation of virtual cities is a very important subject of development in virtual reality applications, since they can serve multiple activities such as entertainment (games, movies..), urban planning, training and education, among others. The size and complexity of such environments, however, makes their modeling through manual approaches (i.e. using software like Sketchup, Blender or Maya) a very expensive and time-consuming endeavor. For example, the city in GTA IV (which is heavily based on New York City) took 3 years to create using such means.

Fortunately, urban environments conform to certain construction directives, which can be encoded into computer algorithms that handle their pattern-repetitive nature. In other words, we can define 3D models as a sequence of programming instructions or 'procedures' - hence the name *procedural modeling* to describe such approaches. A variety of techniques already exist for generating many urban environment features, e.g. terrains [1], roads [2], vegetation [3], buildings [4]. The problem, however, lies on how to define, integrate and control all these techniques, a process that is not only unattractive, but also very hard to grasp and manipulate by artists without a programming or scripting background.

A Graph-based Approach

The alternative I've been working on consists in using interactive graphs [5], a representation that has become increasingly popular among various artists tools. In this case, the nodes represent modeling *operations* and the edges describe the *data flow* from one node to the next (see Figure 1). The connections are established between input and output ports that are anchored to nodes. The data that is passed between nodes is organized into sets of vertices, edges and faces. Such sets are simply called *shapes* [5].

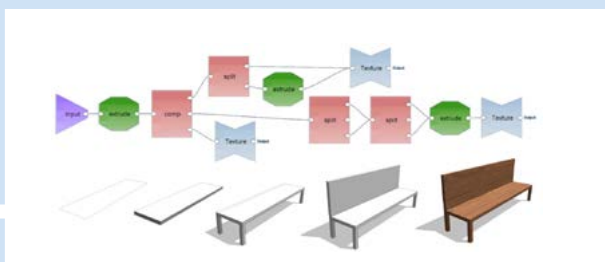


Figure 1: Graph for the generation of a textured bench model. The model examples on the bottom illustrate some of the intermediate steps of the generation process in which a simple shape is progressively evolved into a more complex one.

Once a node receives a shape in its input ports, it transforms, copies or splits the shape and sends out one or more resulting shapes through its output port(s). The modeling process typically starts with a simple shape (for example, a rectangle), and then progressively evolves it by applying a sequence of such operation nodes. The final model consists of all shapes that emerge from unconnected ports.

As shown in Figure 1, each input port accepts multiple incoming edges so as to allow the convergence of certain operation sequences. Output ports can also be the source of multiple outgoing edges, in which case shapes are copied and sent in different directions. It is also possible to create recursive loops within the graph by connecting an output port of a child node to the input port of a parent node.

In addition to the geometric data flow, this visual paradigm also represents the parametric data flow across the graph nodes, which is fundamental to steer the modeling processes. In the graph properties, one can specify *custom shape attributes* (e.g. a string, a number, a boolean) that can be read at any point. Each shape holds a copy, which can be modified. These changes are propagated along the graph: for example, after a split or copy operation, resulting shapes inherit the attributes of their parents.

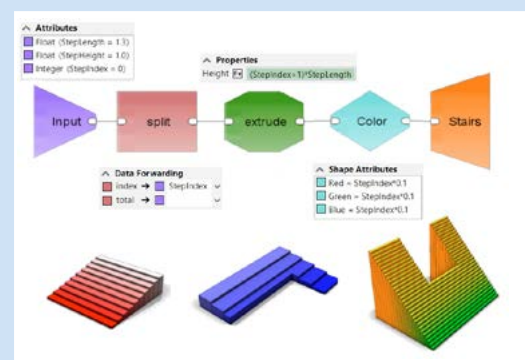


Figure 2: A simple graph example for building a stairs model. The 3 models on the bottom are created using the same graph, but using different parameters and starting shapes.

Each node and edge offers a different set of configuration *parameters*. For example, a "Color" node, which paints a shape with a certain color, has 3 editable parameters: the Red, Green and Blue components. Besides simple constant values, any of these parameters can also be expressed using mathematical functions on some attributes of the manipulated shape. The graph in Figure 2 indicates that we are taking as input any kind of shape and splitting it into multiple shapes of a certain size, as defined in the "Split" node. After the split, an attribute "StepIndex" is updated with a sequential index number, which is then used to give each step a different height and color. Albeit a simple example, this sequence of 'instructions' can produce various results by either changing the initial shape (from a rectangle to an L-shape or U-shape) or some node parameters (the height of the step or the base color).



Building Components

Once assembled, an entire graph, like that of the stairs in Figure 2, can be collapsed into a *component* so as to be used as a node in other graphs. This is demonstrated in Figure 3, together with other component nodes that already address the construction of other building parts. By combining such nodes properly, one can very easily assemble different houses.

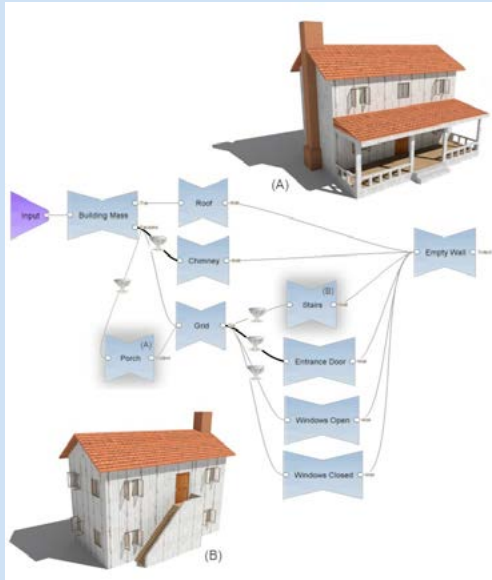


Figure 3: A graph featuring the combination of nodes, each addressing the construction of a higher-level architectural object.

While components can be used to simply encapsulate small routines without specific meaning, they can also be used to build architecturally significant elements, such as building parts, façade styles or windows. Considering that each node and port has a certain architectural meaning, the whole graph becomes a structure of higher semantic concepts, instead of mere geometric operations.

The two models in Figure 3 share almost the same graph specification with only slight differences (highlighted nodes and edges). The “Porch” node exists only in the graph of model A, while the “Stairs” only in that of B. Other differences include the locations of the chimney and entrance door on the façade. Other properties, such as building height, materials, colors, etc. could also be changed by manipulating other node parameters.

Building Cities

Generating cities follows the same concept. By collapsing the example of Figure 3 into a node, and combining it with further component nodes that address, for example, terrain, street and tree generation, one can let the computer create ‘infinite’ cities. To produce variation in the parameters, the strategy lies on using a certain degree of randomness: let the computer pick between X and Y for the building height, between M and N to decide the number of branches in a tree, between V and W to choose the color of the wall, etc.

While this approach is effective for generating realistic-looking cities, it is not adequate for reproducing real cities (such as the New York example of GTA IV). Instead, one can rely on nodes that load information from external sources, such as photographs, maps or any other sort of urban databases, such as those typically owned by municipalities. This information can be saved into attributes, which in turn are used as other node parameters instead of random values.

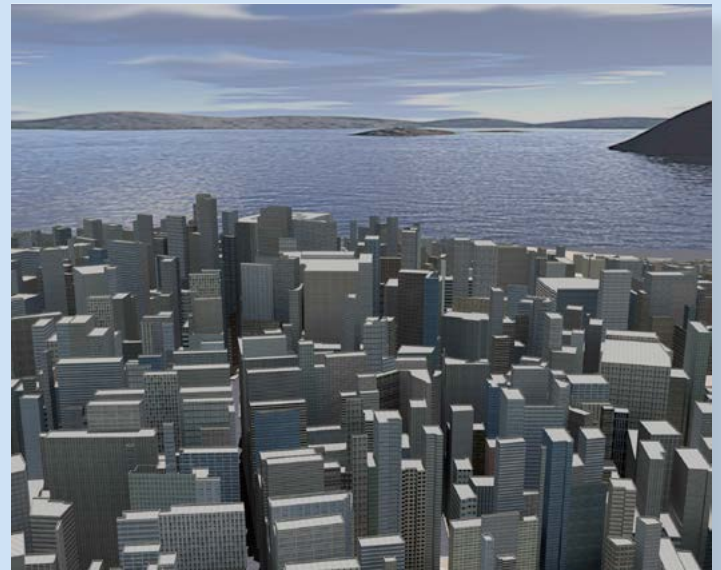


Figure 4: A procedurally modeled city using texture variations of building, street and block components.

Conclusion

In this article I have briefly tackled the topic of procedural modeling in the scope of large city generation. In order to make such a technique accessible to artists, I have worked on a graph-based representation that allows the definition of modeling instructions, which are versatile enough to very easily produce variations of a model. When collapsed into components, one can operate on a higher level of abstraction, expressing concepts such as “Stairs” and “House” instead of “Split” and “Extrude”. By either using random values or loading them from external sources, the same graphs can be used to produce distinct results, ultimately generating city models with unlimited variety of objects.

Procedural modeling is still a rather novel research topic and there is still much to be done to improve control and user experience. In addition to the generation of geometry, I have been also focusing on other entities that compose an interactive and animated virtual world: cars that move on the roads, street lights that can be toggled on and off, doors that can be opened and closed, etc. They could be easily integrated in the whole procedural process.

If you would like to learn more about this work, please refer to [5] or contact me at pedro.brandao.silva@gmail.com.

References

- [1] Smelik, R. M., de Kraker, K. J., Tutenel, T., Bidarra, R., & Groenewegen, S. A., *A survey of procedural methods for terrain modelling*, 2009.
- [2] Galin, E., Peytavie, A., Guérin, E., & Beneš, B., *Authoring Hierarchical Road Networks*, Computer Graphics Forum, 30(7), 2021–2030, 2011.
- [3] Prusinkiewicz, P., & Lindenmayer, A., *The Algorithmic Beauty of Plants*, Springer-Verlag, 1996.
- [4] Müller, P., Wonka, P., Haegler, S., Ulmer, A. & Van Gool, L., & Gool, L. Van., *Procedural Modeling of Buildings*, ACM SIGGRAPH 2006 Papers (Vol. 25, pp. 614–623). Boston, Massachusetts: ACM, 2006.
- [5] Silva, P. B., Müller, P., Bidarra, R., & Coelho, A., *Node-based Shape Grammar Representation and Editing*, In Proceedings of the 2013 Workshop on Procedural Content Generation in Games - PCG '13, 2013.



Dynamic Peer-to-Peer Game Networks using WebRTC

Joost Verdoorn

WebRTC (Web Real-Time Communication) is a set of promising new web technologies that aim to bring the benefits of peer-to-peer communication to the web browser. It allows web browsers to directly communicate with one another, without the communication having to be routed through a third party, e.g. a central server. With the help of these peer-to-peer connections, developers will be able to build an array of completely new web applications, such as bittorrent clients and voice-over-IP applications, to be run right in the web browser. WebRTC seems to be yet another omen that more and more applications that once lived on the desktop will be moving to the web browser.

Last summer Karens Grigorjancs, Jasper Abbink and myself conducted our bachelor project on WebRTC. Our goal was to reach for the technical limits of WebRTC and develop a library that enables developers to easily set up a dynamic peer-to-peer game network for browser-based multiplayer games, and to create a small browser-based multiplayer game as an entertaining way to demonstrate the features of this library. Throughout the summer we worked on-site at TNO, our supervisor and client.

TNO is a non-profit research organisation with a focus on applied sciences, and has always shown a strong interest in communication and the internet. As such, TNO keeps an eye out for new technologies that might help shape the future of communication. TNO pointed us in the direction of WebRTC when the three of us approached them with the offer to do our bachelor project with them, and after reading up on the technology, our interest was piqued.

WebRTC

WebRTC has been in development for almost three years now. It first saw the light of day when in January 2011 Ericsson Labs created a demonstration of a peer-to-peer video chat application that was run right in the web browser. Google soon picked up the project, likely out of a need for richer applications that can run in the web browser to push their Chromebook line of laptops. The W3C and IETF standards organisations took up the task of defining the official specifications and deciding which protocols to use. Since then, the project has evolved to the point that there are now mature implementations of the specifications in both Firefox and Chrome, and a few less popular web browsers.

The initial focus of WebRTC was on video and audio chat functionality, and as such, these were the features that were first implemented. However, it

was soon found that if there were a way to send raw data in WebRTC it would benefit web developers even more. Now that video and audio functionality was nearly done, focus shifted towards the development of the data channel API. Because work on raw data was started some time later than video or audio streams, data channels are not as stable yet, but we will tell more about this later.

The three of us soon agreed that we would like to work on a marriage of browser-based gaming and WebRTC. WebRTC's benefits for gaming are clear: it provides direct communication between two browsers, and this means that we could achieve both low latencies and high data throughput without having to hire expensive servers to guide this traffic. Primarily the former benefit was interesting to us, as even any casual gamer knows: low latencies are very important in most kinds of multiplayer gaming. Before even having started on the bachelor project we created a first tech demo where we used a mobile device running the Chrome browser to control a game of Pong on the desktop, where the mobile's G-Sensor was used to control the pad in game.

Having used this tech demo to orient ourselves in the field, we wrote our bachelor project proposal in which we proposed to build our own a browser-based game for which we would use the mobile as input device, and making it a much richer coupling than we did in the Pong game. TNO rejected this proposal, stating that it would not really push the boundaries of WebRTC much, and it would not be much of a challenge as we had already implemented the mobile-to-desktop WebRTC communication in our tech demo. After we gave it much thought, we proposed to develop a library for setting up a dynamic peer-to-peer game network, connecting a moderate amount of players to each other, and TNO agreed. In the end we are glad that TNO rejected our first proposal, as we found great joy in solving the puzzles of creating a large scale network.

The Network

Different types of multiplayer games have very different requirements when it comes to their networks. Role playing games such as World of Warcraft can deal with a higher latency but require a great amount of simultaneous connections, while in a fast-paced action game like Counter Strike a very low latency is necessary. Before we started building the library, we would need to decide what type of network we required, and thus what type of game we would develop. Eventually we decided to build a small arcade action game, as low latencies are one of WebRTC's great benefits and we wanted to put it to good use. One of the requirements of the network was that the game could be played with a fair amount of players.

When only a few peers make up a peer-to-peer network, it poses no problem to make sure all traffic has a low latency given a decent broadband connection. One can just hook up every peer to every other peer to ensure the shortest connections between them. When one peer wants to talk to another, he will always have a direct communication channel to that peer, and so there is no



need for any routing of packages between the peers. When more and more peers join the network, however, further steps have to be taken to ensure decent latencies. Maintaining a large number of open connections means a lot of overhead for every peer. Apart from that, it also requires peers to duplicate messages a great number of times if they want to broadcast it to all other peers, which requires a high amount of bandwidth. Ultimately this way of setting up a network does not scale very well - it needs some structure.

Fortunately, a lot of research has gone into large-scale peer-to-peer networks since the start of the Information Age. We read quite a few papers on the subject and combined techniques from some of them to build our network. We used an algorithm called Vivaldi to arrange peers in an n -dimensional Euclidean space, where peers with a lower latency between them will be arranged closer to each other, and those with a higher latency to be further apart. Building on top of this, the PoPCorn algorithm will elect super peers which will take on the task of routing messages between peers. This means that, in order to talk to each other, two peers no longer have to be directly connected, but are able to communicate via connected super peers.

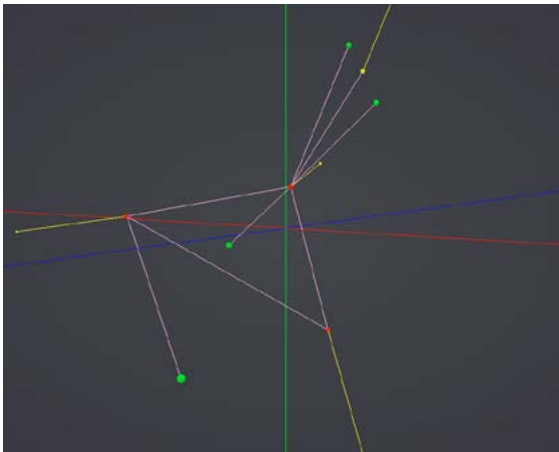


Figure 1: Connected peers are arranged in 3-dimensional Euclidean space. Red dots represent super peers, green dots represent normal peers, and yellow dots represent candidate super peers.

Implementing these algorithms took most of our time. We soon found out that we had to cover all sorts of unexpected behaviour, such as super peers randomly dropping from the network - sometimes causing what is known as a net-split - and delayed communications between peers. We had to build in fail-safes and sanity checks to maintain the network's integrity. One thing we deliberately did not work on was protecting the network from malintent, which would have justified a bachelor project of its own.

Not Ready Yet

At times, it was clear that the WebRTC implementations were still under heavy development. Although Chrome and Firefox are able to communicate with each other, this communication was severely limited at the time, allowing only video and audio streams but no raw data to be exchanged. Apart from that, both browsers implement a different API, where neither of them completely followed the official specifications. As the browsers were incompatible for our purposes, we chose to develop our library primarily for Chrome, as it came closest to following the official specs.

However, Chrome's implementation of the WebRTC specifications is far from perfect. It is poorly documented which part of the official specifications it

follows. For example, it turned out that by default Chrome's raw data channels are rate limited to a mere 30 kilobits per second, which is pretty much useless for any serious networking. When attempting to exchange data at a higher rate, Chrome would throw indelible errors. It turned out this rate could quite easily be increased, but in a very hacky way: we had to do a string-replace on a so-called Session Description package that would be sent when establishing the connection, in which the data rate was hard-coded.

It is clear though that despite its current flaws, WebRTC is gaining a lot of traction among developers. When we started off, barely any information was published on the technology, let alone implementations of it in web applications. At the end of the summer, the internet was already teeming with libraries and apps that made great use of it, ranging from simple consumer support services through video and audio to a fully-functional torrent client. It seems that WebRTC has a very bright future.



Figure 2: A screenshot of the finished game. Players fly around in UFOs and can fight each other with cannons.

Our bachelor project was a success as well. Even though we had no experience with developing either games or peer-to-peer networks, both turned out to be a lot of fun. The fruits of our labors are distributed under the open-source BSD license, and the game can be played at <http://webrtc.jstfy.com>.



Java Puzzlers

De Puzzler

Last MaCHazine you had the puzzle 'Inclement Increment'. Several solutions have been submitted. The best and most complete answer is submitted by Laurent Verweijen. Now we present the solution of this puzzle.

At first glance, the program might appear to print `j` 100. After all, it does increment `j` times. Perhaps suprisingly, it does not print 100 but 0. All that incrementing gets us nowhere. Why?

As the puzzle's title suggests, the problem lies in the statement that does the increment:

```
j = j++;
```

Presumably, the author of the statement meant for it to add 1 to the value of `j`, which is what the expression `j++` does. Unfortunately, the author inadvertently assigned the value of this expression back to `j`. When placed after a variable, the `++` operator functions as the *postfix increment operator* [JLS 15.14.2]: The value of the expression `j++` is the original value of `j` before it was incremented. Therefore, the preceding assignment first saves the value of `j`, then sets `j` to its value plus 1, and, finally, resets `j` back to its original value. In other words, the assignment is equivalent to this sequence of statements:

```
int tmp = j;  
j = j + 1;  
j = tmp;
```

The program repeats this process 100 times, after which the value of `j` is exactly what it was before the loop, or 0.

Fixing the program is as simple as removing the extraneous assignment from the loop, leaving:

```
for (int i = 0; i < 100; i++)  
    j++;
```

With this modification, the program prints 100 as expected.

The lesson is: **Do not assign to the same variable more than once in a single expression.** An expression containing multiple assignments to the same variable is confusing and seldom does what you want.

Each MaCHazine there will be a puzzle with a script of a program, here is the new one. It is up to you to find out what the program does. You may use Java, but it's more fun just to use your mind.

The following program simulates a simple clock. Its loop variable represents a millisecond counter that goes from 0 to the number of milliseconds in an hour. The body of the loop increments a minute counter at regular intervals. Finally, the program prints the minute counter. What does it print?

```
public class Clock {  
    public static void main(String[] args) {  
        int minutes = 0;  
        for (int ms = 0; ms < 60*60*1000; ms++)  
            if (ms % 60*1000 == 0)  
                minutes++;  
        System.out.println(minutes);  
    }  
}
```

References

- [1] Java puzzlers, traps Pitfalls and corner cases, Joshua Bloch & Neal Gafter
- [2] The Java® Language Specification, James Gosling, Bill Joy, Guy Steele, Gilad Bracha & Alex Buckley

Stuur je oplossing naar machazine@ch.tudelft.nl
en maak kans op een prijs t.w.v. 45 euro!

Stuur niet alleen je antwoord op, maar ook een
uitleg hoe je tot dit antwoord gekomen bent.



Wiskundepuzzel

Mark Veraar

Nieuwe opgave

Laat $f, g : [0, \infty) \rightarrow (0, \infty)$ continue functies zijn en definieer $h : (0, \infty) \rightarrow (0, \infty)$ door

$$h(x) = \frac{\int_0^x f(t) dx}{\int_0^x g(t) dx}$$

Bewijs het volgende: als de functie $\frac{f}{g}$ dalend is, dan is h ook dalend.

Vorige opgave met uitwerking

We spreken af $\mathbb{N} = \{1, 2, \dots\}$.

Construeer een functie $f : \mathbb{N} \rightarrow \mathbb{N}$ zodat $f(f(n)) = n^2$ voor alle $n \in \mathbb{N}$.

Er zijn inzendingen ontvangen van Jan Aarts, Dirk Hazenoot, Emiel Lorst, en John Simons. Er zijn 2 mogelijke uitwerkingen. De uitwerkingen van Jan Aarts, Dirk Hazenoot, en John Simons berusten op priemfactorisatie en de uitwerking van Emiel Lorst berust op ordening. Hieronder het idee van beide oplossingen. Laten we in elk geval alvast definiëren $f(1) = 1$.

Oplossing 1: Laat $(p_n)_{n \geq 1}$ een aftelling van de priemgetallen zijn. Definieer $f(p_{2j-1}^k) = p_{2j}^k$ en $f(p_{2j}^k) = p_{2j-1}^k$ voor $j, k \in \mathbb{N}$. Met deze afspraak is $f(f(p_j^k)) = p_j^k$ voor $j, k \in \mathbb{N}$. Zet deze definitie voort door de afspraak dat de functie multiplicatief is: $f(ab) = f(a)f(b)$, voor alle $a, b \in \mathbb{N}$. Nu is het eenvoudig om na te gaan dat $f(f(n)) = n^2$ voor alle $n \in \mathbb{N}$.

Oplossing 2: Laat $V = \{n^2 : n \in \mathbb{N} \setminus \{0, 1\}\}$. We definiëren eerst $f(j)$ voor alle $j \geq 2$ en $j \notin V$. Zo'n element kunnen we schrijven als $n^2 + k$ met $1 \leq k \leq 2n$. Merk op dat geen van de getallen $n^2 + k$ met $1 \leq k \leq 2n$ een kwadraat is. Definieer voor alle $1 \leq k \leq 2n$

$$f(n^2 + k) = \begin{cases} n^2 + k + 1, & \text{als } k \text{ is oneven,} \\ (n^2 + k - 1)^2, & \text{als } k \text{ is even.} \end{cases}$$

Nu is duidelijk dat voor alle $1 \leq k \leq 2n$ oneven geldt $f(f(n^2 + k)) = f(n^2 + k + 1) = (n^2 + k)^2$. Merk op dat $n^2 + k + 1$ voor $1 \leq k \leq 2n$ oneven nooit een kwadraat is. Definieer f voor de getallen uit V recursief als $f(n^2) = f(n)^2$. Dit is goed gedefinieerd omdat voor alle $n \geq 2$ geldt $n^2 > n$. We gaan na dat ook alle $j = n^2 \in V$ aan de eis voldoen. Inderdaad, $f(f(n^2)) = f(f(n)^2) = f(f(n))^2 = (n^2)^2$. Ten slotte merken we op dat door deze definitie ook voor $1 \leq k \leq 2n$ even geldt $f(f(n^2 + k)) = f((n^2 + k - 1)^2) = f(n^2 + k - 1)^2 = (n^2 + k)^2$. 

Wiskundeladder

1.	Prof. Dr. JM Aarts	10
	Emiel Lorst	10
2.	Jeroen Wille	9
3.	Marieke van der Tuin	8
	Erik Ammerlaan	8
	Robbie Elbertse	8
4.	Micke Vrolijk	6
	Leon Planken	6
	Jonatan Bijl	6
5.	Jarno Hartog	5
	Jeff Smits	5
	Michiel de Reus	5
	Wim van Geloven	5
	M. Looijen	5
	Romke Rozendaal	5
	Jolien de Haas	5
	Sander Gribbling	5
	Dirk Hazenoot	4
	John Simons	3

The Impact of Mathematical Modeling on the Production of Special Purpose Cement

Domenico Lahaye

Introduction

What impact can mathematical modeling have in the production industry? Our partnership with Almatris B.V., a special purpose cement manufacturer, resulted in the complete elimination of unscheduled plant shut-downs. Almatris now reports a much more stable manufacturing process and hence a very significant productivity increase. The key to our success resides in the deployment of advanced computational fluid dynamics techniques. It is currently driving a long-term collaboration expanding into different branches of Almatris. Students employed on the project can train in the mathematical modeling of turbulent combustion, granular material flow and material phase transformations. They are offered the opportunity to experiment with modern simulation software on parallel hardware and to interact with experts in various engineering disciplines.

Production of Special Purpose Cement

Commonly used cement is easy to manufacture. A long cylindrical rotating furnace or *kiln* is typically employed in the process. I have schematically represented the production process in Figure 1. Cement kilns are slightly tilted to facilitate the lateral downward motion of the material. As the kiln rotates about its axis, the powder material traverses the furnace from the inlet to the outlet. It is being mixed and heated up to a sufficiently high temperature. These processes cause the reactions that transform the raw materials into the final product.

However, the special purpose cement produced by Almatris (www.almatris.com) requires processing the mixture of a calcium-bearing limestone and an aluminum-bearing material at high temperature. This requires the control of the critical process parameters such as the temperature inside the oven and its rotational speed. Experience at the Almatris plant in Rotterdam has shown that neither trial and error nor empirical modeling provide sufficient insight into these critical conditions. Measuring the temperature inside the kiln has proven to be difficult due to the harsh operating conditions. Without accurate temperature readings, the amount of heat absorbed by the material remains hard to track. The stringent product quality specifications imposed by Almatris are therefore hard to attain. The efficiency of the kiln is doomed to remain suboptimal. As a world-leader in calcium-alumina special purpose cement, Almatris is eager to invest in research partnership and to use cutting edge technologies. This will allow to better serve various industries that rely on Almatris cement in applications exposed to high temperature.

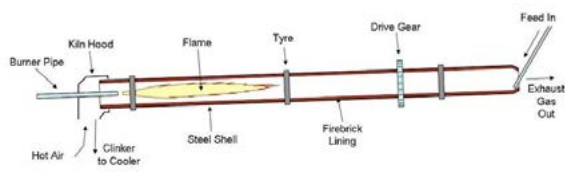


Figure 1: General layout of a rotary kiln used at Almatris and modeled by the scientific computing group.

The scientific computing group developed a detailed mathematical model that accurately predicts the temperature, the radiative heat distribution, and the absorption of heat by the material bed inside the oven. The use of advanced computational fluid dynamics techniques proved to be vital. In the next sections, I will describe the mathematical models and the results obtained in some more detail.

Mathematical Modeling

Michele Pisaroni and Miguel Romero built the mathematical model of the rotary kiln in use by Almatris at their plant in Rotterdam during their master degree. They executed the project in two stages. In the first stage, Michele relied on the fact that only a small fraction of the space inside the kiln is taken up by cement material. This allowed him to assume the kiln to be empty and to develop a three-dimensional non-premixed turbulent combustion model that predicts the temperature and radiative heat profile distribution inside the kiln. In the second stage Miguel built upon Michele's results and constructed a system of coupled ordinary differential equations for the absorption of heat by the material as it traverses through oven and for the ensuing phase transformation and chemical reactions. The combination of the two models provided Almatris with a better grip on its production process.

Modeling Turbulent Combustion

Combustion [3] is the physical process in which fuel and air mix and chemically react. This reaction produces heat and combustion products such as water and carbon-monoxide. Describing combustion requires capturing both the flow phenomena and the chemistry involved. In our case, the combustion is fed by an inflow of natural gas and air through distinct channels in the burner pipe shown to the far left of Figure 1. Fuel and air reach the combustion area prior to being mixed. The amount of inflow of fuel and air and heat generated cause the flow to quickly exceed the laminar regime and to become turbulent. The combustion is said to be a non-premixed turbulent combustion.

Turbulent combustion is difficult to model due to its chaotic flow pattern. A sufficiently accurate description is, however, indispensable in quantifying the mixing of fuel and air as well as the intensity of the chemical reactions. The occurrence of turbulence, in particular, enhances the mixing. Adding to the difficulty is the fact that the heat released by the chemical reactions, in turn, strongly affects the turbulent flow.

Numerical simulations environments capture turbulent combustion by imposing the conservation of mass, momentum and energy of the individual species in the reacting flow. This leads to the set of Navier-Stokes equations to which a model for the combustion and the turbulence is added. These set of partial differential equations is discretized by a finite volume method on an unstructured polyhedral mesh and solved by multigrid preconditioned Krylov methods.

A sample result of the computations by Michele Pisaroni is given in Figure 2. This figure shows the temperature along the axis of the kiln for two operating conditions differ in the amount of air used. The increase of the amount of air reduces the expected peak temperature. The location of the peak temperature and the amount by which it is reduced could, however, only be revealed through numerical simulation.



Modeling Material Processing

Miguel Romero constructed a mathematical model for the processing of the granular materials inside the kiln. He first quantified how the rotary motion of the kiln about its inclined axis causes these materials to mix and to flow in axial direction. To this end, he studied both discrete element methods [4] as well as extensions of the Navier-Stokes equations that takes two phases (solid and air) into account. Next, he identified the different paths and mechanism for the heating up of the materials. Both the shell surrounding the kiln and the air-gas mixture flowing inside it transmit heat to the materials via diffusion, convection and radiation. In the final stage Miguel linked the temperature of the mixture of materials to the chemical reactions and to the phase transformations. The model he constructed predicts the position, temperature and amount of liquid fraction of an infinitesimal amount of mixture of the granular material as it traverses the kiln. This amount of liquid fraction is a reliable indicator of the quality of the final cement product. The model requires as input the temperature and the radiative heat distribution computed by Michele at an earlier stage.

Miguel's model explains how the difference in the temperature profiles shown in Figure 2 affects the liquid fraction and therefore the quality of the end product. Data recorded at the Almatris plant in Rotterdam has repeatedly shown that the model makes accurate predictions. Such data is shown in Figure 3. This figure shows how the produced amount of material A, B and C meeting the stringent quality requirement has significantly increased after changing the temperature profile.

During the course of their work Michele and Miguel had ample opportunities to interact with and to learn from leading experts in combustion and granular flow. Michele travelled to London for a training in the simulation software he has been using, while Miguel extensively discussed his work with a senior scientist at the CSIRO in Camberra, Australia. Their work resulted in numerous conference contributions and two journal articles [2, 5]. A video highlighting the collaboration between the scientific computing group and Almatris is available at the website [1]. Michele is currently pursuing a PhD at the EPFL in Lausanne, and Miguel holds a research position at BASF in Ludwigshafen.

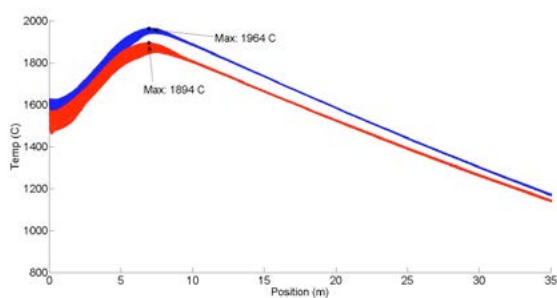


Figure 2: Computed temperature profiles along the axis of the kiln for two operating conditions.

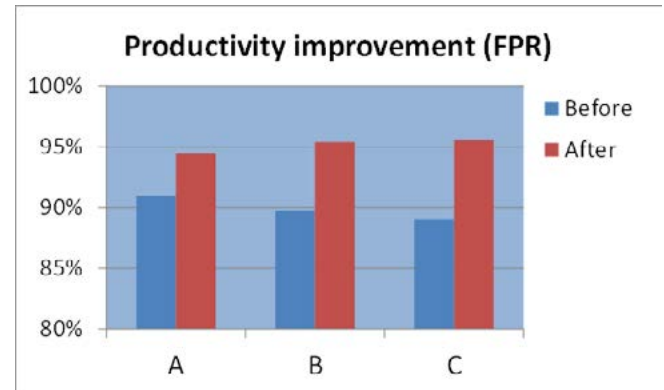


Figure 3: Measured productivity increase for the products A, B and C.

Future Challenges and Opportunities

The impact of the work of Michele and Miguel is driving Almatris to further invest in the collaboration with the scientific computing group. Many questions on the operation of the rotary kiln remain unanswered. Almatris is eager to invest in more research on combustion to render the process more fuel efficient and more environmental friendly. Further research into the mixing of the granular material processing is indispensable in meeting the ever growing demands of the customer. Almatris is looking for a PhD student willing to take up these challenges.

Research has furthermore, expanded into the modeling of so-called vertical shaft kilns operated by Almatris. In these kilns the reinforcement for the high quality cement is produced. Two students, Lu Cheng and Bonnie Fan, have worked on the combustion modeling and the material processing in these kilns, respectively. Unlike the cement kiln, this kiln is completely filled by the material and the two processes cannot be decoupled. Almatris has student internships available to model the shaft kiln in a fully coupled manner.

The partnership between Almatris and the scientific computing group is awaiting a bright future and invites motivated students to contribute to the success story.

References

- [1] D. J. P. Lahaye. Almatris TU Delft promotion video: www.youtube.com/watch?v=axifpqolpmq.
- [2] M. Pisaroni, R. Sadi, and D. Lahaye. Counteracting ring formation in rotary kilns. *Journal of Mathematics in Industry*, 2(1):1–19, 2012.
- [3] T. Poinot and D. Veynante. *Theoretical and Numerical Combustion*. R.T. Edwards, Inc., second edition, 2005.
- [4] K. K. Rao and P. R. Nott. *An Introduction to Granular Flow*. Cambridge Series in Chemical Engineering. Cambridge University Press, 2008.
- [5] M. A. Romero-Valle, M. Pisaroni, D. Van Puyvelde, D. J. P. Lahaye, and R. Sadi. Numerical modeling of rotary kiln productivity increase. Technical Report 13-09, Department of Applied Mathematics, TU Delft, 2013.



Noise minimization on houses around airports

Ir. Teun Janssen

Aviation has impact on the environment. One of these forms of impact it is, is the noise pollution suffered by people living in the vicinity of an airport.

My thesis was a combined project of the CWI (the national research institute for mathematics and computer science in the Netherlands) and the NLR (National Aerospace Laboratory of the Netherlands). In the thesis we examine noise pollution around airports. Given an airport we consider the problem of maximizing the number of flights, while minimizing the number of houses suffering more than the threshold amount of noise pollution. We formulate this problem as a multi-objective optimization problem and look at the computational complexity of the problem and its approximability. Using the concept of Pareto optimality we consider methods to rewrite our problem to a single objective optimization problem. Using the methods and the computational complexity and approximability results obtained, we construct algorithms to solve the problem. Finally we use these algorithms on problem instances to give insights in their performance.

In this article we will focus on one of the two models used and we will look at the most important complexity result obtained in the thesis.

The model

We consider an airport. For this airport we are given the location of its runways and the locations of houses in its neighborhood. Furthermore we are given a collection of flight routines. Each flight routine is a unique combination of the following:

- The aircraft type associated with the flight movement.
- The runway used and in what direction.
- The flight procedure the aircraft is following (which incorporates its ascend or descend, its speed profile and the accompanying thrust settings.)
- The flight trajectory it follows, i.e., the coordinates a flight is traversing in the plane.

Now for every single flight routine and every housing location we are given an amount of *noise pollution* that this flight routine confers to that housing location. For the noise pollution on the houses we are given a threshold. This threshold represent the maximum amount of noise pollution, that is acceptable.

Around the airport we are also given a zone imposed by government legislations. Outside this zone the noise pollutions may not exceed a maximum legislated level. The government enforces this legislation by a set of measure points in the neighborhood of this zone. Since these measure points will not always be on the zone, maximum levels in the measure points may differ from the government legislated level. Given these, we will consider the problem of maximizing the number of flights, while minimizing the number of houses suffering more than the threshold amount of noise pollution.

We will now formulate this problem as a multi-objective optimization problem. We will call it the *route scheduling problem* (RS).



For an instance I_{RS} of the route scheduling problem, we have the following sets given

Flight routines	$\{r_1, \dots, r_m\} = R$
Measure points	$\{q_1, \dots, q_l\} = Q$
Housing locations	$\{h_1, \dots, h_n\} = H$

Furthermore we have the following given constants

$\alpha_{qr} \in \mathbb{R}_+$	$\forall q \in Q, \forall r \in R$	Representing the noise pollution of flight routine r on measure point q
$\beta_{hr} \in \mathbb{R}_+$	$\forall h \in H, \forall r \in R$	Representing the noise pollution of flight routine r on house h
$\omega_h \in \mathbb{R}_+$	$\forall h \in H$	Representing the number of houses on every housing location h
$\theta \in \mathbb{R}_+$		The maximum noise allowed by the government
$\lambda \in \mathbb{R}_+$		The maximum noise allowed on every house

An instance I_{RS} will contain the above constants and sets. Now we introduce the set of variables $x = \{x_{r_1}, \dots, x_{r_m}\}$ which represent the amount of times a routine r is used with $x_r \in \mathbb{Z}_+, \forall r \in R$. In our problem we have the following two objective functions

$$\max \sum_{r \in R} x_r \quad \text{and} \quad \min \sum_{h \in H} \omega_h z_h.$$

Maximum feasible subsystem formulation

We will formulate the problem as a *maximum feasible subsystem* problem (MFS). In MFS we are given a set of linear relations and we want to find a solution that satisfies the most of these relations. A relation can be of the type $=, \geq, >$ or $\neq$. More formally one is given a linear system $Ax \diamond b$, with $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$ and with $\diamond = \{=, \geq, >, \neq\}$, where one wants to find an $x \in \mathbb{R}^m$ that satisfies the most linear equations.



In addition to the set of optional relations $Ax \diamond b$, one sometimes also has a set of binding relations $Cx \diamond d$ that have to be satisfied, with $C \in \mathbb{R}^{l \times m}$, $d \in \mathbb{R}^l$. This version of MFS is called the constrained maximum feasible subsystem problem. Furthermore sometimes one requires x to be binary or integer instead of real valued. Lastly one can also consider a weighted version of MFS, where all optional constraints have a certain weight w_i assigned to them for all $i \in \{1, \dots, n\}$. One then maximizes the weight of the satisfied constraints instead of the number of satisfied constraints.

Since MFS only has one objective function, we need to get rid of one of the objective functions of RS. We will do this by setting a lower bound on the number of flights.

$$\sum_{r \in R} x_r \geq \kappa$$

Bounding a objective value in such a way is known as the *bounded objective value method* which is used on multi-objective optimization problems to convert them to single objective problems and it is used to obtain Pareto optimal solutions [2].

We will proceed by writing our problem as a weighted constrained integer maximum feasible subsystem problem. We consider the following system of constraints

$$\begin{aligned} \sum_{r \in R} \beta_{hr} x_r &\leq \lambda & \forall h \in H & \text{ (optional constraints)} \\ \sum_{r \in R} \alpha_{qr} x_r &\leq \theta & \forall q \in Q & \text{ (binding constraints)} \\ \sum_{r \in R} x_r &\geq \kappa & & \text{ (binding constraint)} \\ x_r &\in \mathbb{Z}_+, & \forall r \in R & \text{ (binding constraints)} \end{aligned} \quad (1)$$

Thus we have a system of n optional constraints and $l + n + 1$ binding constraints. Lastly we assign weights to our optional constraints, i.e., $w_h = \omega_h, \forall h \in H$. Note that sometimes we assume without loss of generality that $x_r \in \{0, 1\}, \forall r \in R$. In this case we assume that we have enough duplicates of every flight movement in our set R such that we can take any reasonable amount of every flight movement, i.e., we consider R as a multiset.

Inapproximability

In the thesis it is proven that the problem is $\mathcal{NP}$ -complete [1]. However a more striking result obtained is the following theorem.

Theorem 1. *The problem route scheduling problem is as hard to approximate as maximum independent set.*

Proof. We will prove these results by doing a cost preserving polynomial time reduction to the *maximum independent set* problem (MIS). In graph form MIS is the following problem: Given a (hyper)-graph $G(V, E)$, where V are the vertices and E the edges, one wants to find a subset of vertices V' such that no two vertices in V' share an edge.

Let $G(V, E)$ be a graph of an arbitrary instance of MIS and let $|V| = \eta$ be the cardinality of V . For every node $v \in V$ let $N(v)$ denote the nodes incident to v , i.e., $N(v) := \{u \in V : \{u, v\} \in E\}$. We then consider the following $|V|$ inequalities

$$\begin{aligned} -x_v + \sum_{u \in N(v)} x_u &\leq -1 & \forall v \in V & \text{ (Optional constraints)} \\ x_v &\in \{0, 1\} & \forall v \in V & \end{aligned}$$

We see that we satisfy the inequality for v if and only if $x_v = 1$ and $x_u = 0, \forall u \in N(v)$. Thus if we satisfy s inequalities, we obtain an independent set of size s by taking all vertices v for which $x_v = 1$. Conversely if we have an independent set I of size s we can satisfy s inequalities by setting $x_v = 1$ if $v \in I$ and setting $x_v = 0$ if $v \in V \setminus I$.

Now consider the following instance of RS. Let $R = V \cup \bar{V} = \{v_1, \dots, v_\eta\} \cup$

$\{\bar{v}_1, \dots, \bar{v}_\eta\}$, $H = V$, $\kappa = |V|$ and $\lambda = |V| - 1$. Note that $|R| = 2|V|$ and $|H| = |V|$. Furthermore we take β_{hr} as follows for every $v \in H$

$$\beta_{vr} = \begin{cases} 0 & r = v \\ 2 & r = u \in N(v) \\ 1 & r = u \in V \setminus (N(v) \cup v) \\ 1 & r = \bar{u} \in \bar{V} \end{cases} \quad (2)$$

Then we have the following set of optional constraints

$$0x_v + \sum_{r \in N(v)} 2x_r + \sum_{r \in V \setminus (N(v) \cup v)} x_r + \sum_{r \in \bar{V}} x_r \leq \lambda, \quad \forall h = v \in H$$

with $x_r \in \{0, 1\}, \forall r \in R$. We can assume that $\sum_{r \in R} x_r = \kappa$. This is due to the fact that we can only satisfy an optional equation if $\sum_{r \in R} x_r = \kappa$. To see this, suppose that $\sum_{r \in R} x_r = \kappa + 1$. Then

$$0x_v + \sum_{r \in N(v)} 2x_r + \sum_{r \in V \setminus (N(v) \cup v)} x_r + \sum_{r \in \bar{V}} x_r \geq \kappa > \lambda$$

Thus we can substitute the following equality in our optional constraints

$$\sum_{r \in \bar{V}} x_r = \kappa - \sum_{r \in V} x_r$$

Then we get the following set of optional constraints

$$\begin{aligned} -x_v + \sum_{r \in N(v)} x_r &\leq \lambda - \kappa = -1 & \forall h \in H & \text{ (Optional constraint)} \\ x_r &\in \{0, 1\} & \forall r \in R & \end{aligned}$$

Which is in one to one correspondence to the MIS-problem and can be constructed in polynomial time. Thus we have a cost preserving transformation from MIS to RS.

For every independent set I of size s we can satisfy s inequalities. We do this by setting $x_r = 1$ if $r \in I$ and by setting $\kappa - s$ times an arbitrary $x_r = 1$ where $r \in \bar{V}$. $\square$

Note that this proof also holds for $\beta_{hr} \in \{0, 1, 2\}$, $C \in \mathbb{Z}$ and $\lambda \in \mathbb{Z}$. Since RS is as hard to approximate as maximum independent set, the best approximation algorithm we can hope has an approximation ratio of n , i.e., $\text{ALG} \geq \frac{1}{n} \text{OPT}$, where ALG is the solution obtained by the approximation algorithm and OPT is the optimal solution [3].

Conclusion

Because the route scheduling problem is $\mathcal{NP}$ -complete it is unlikely we can find an efficient algorithm that solves the problem. Furthermore since the problem is as hard to approximate as the maximum independent set problem, the best way we can approximate the problem in polynomial time is by trying to satisfy one constraint, which in general is not a very good approximation. Therefore we will only use two types of algorithms to solve the problem. Exact algorithm that can run in time exponential in the input size or heuristics which run in time polynomial in the input size, but with no performance guarantee.

References

- [1] T. Janssen, *Noise minimization on houses around airports*, Master thesis TU Delft, 2013.
- [2] V. Pareto, *Manuale di economia politica*, volume 13, Societa Editrice 1906.
- [3] D. Zuckerman, *Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number*, Theory of Computing, 3:103-128, 2007.



Mathematics

33



De SuperSlideQam

Waar gaat het heen met de wereld

Technologische trends voor 2014

Bij Q42 maken we apps, web apps en sites voor klanten. Daar zitten een paar flinke tussen, zoals 9292, het Rijksmuseum, de Staatsloterij of Philips. Zulke klanten komen niet alleen bij ons omdat we uitblinken op technisch gebied, maar ook omdat we al vanaf de conceptfase onderdeel van de discussie zijn. We helpen met aanscherpen van het concept, doen de technische validatie en maken vaak direct een werkend HTML prototype als "proof of concept". De opgebouwde kennis en inzichten en de honger van 50 nerds naar al het nieuwe geven ons een aardig idee welke vorm de industrie over een paar jaar zal hebben.

In dit artikel gaan we drie trends toelichten met onze kijk op de zaken. We hopen je zo te inspireren en inzicht te geven in wat er speelt in de wereld van webbureau's en startups.

Home automation, domotica en het Internet of Things

De wereld om je heen op een nieuwe manier beleven door gewone apparaten te verbinden met het internet: dat is de "Internet of Things". Wij denken dat zo'n wereld onvermijdelijk is. Het is immers een logisch gevolg van lagere prijzen, meer toegankelijke hardware en diensten voor developers. De opkomst van de

cloud draagt daar ook aan bij. Door zulke nieuwe mogelijkheden vervaagt de scheidslijn tussen hardware en software en zal die trend zich na early adopters en geeks voortzetten naar de huiskamer. Kijk maar naar 3d printers.

Internet-connected lamps

Je huis en andere zaken in de echte wereld besturen met technologie is een trend waar we direct mee te maken hebben gekregen door ons werk voor Philips Hue (<http://meethue.com>). Hue is een LED-lamp die je kunt besturen met je smartphone. Wij maken de servicelaag die daarbij hoort met behulp van Google App Engine, maar hebben ook een prototype voor een smartphone app ontwikkeld: Hue Limited Edition (<http://goo.gl/iDOWgf>).

Voor Hue zijn we een belangrijke strategische partner geworden, omdat we snel kunnen schakelen met techniek. We haalden de opdracht oorspronkelijk binnen door een HTML prototype van de smartphone app te maken die de product owner intern bij Philips live kon demo-en. En we hebben een goede gut feeling dankzij onze eigen belangstelling voor dit soort technologie. Zo hebben een aantal Q'ers ondertussen een Fitbit (<http://fitbit.com>), waarmee ze hun stappen tellen en de resultaten en analytics daarvan gebruiken om elkaar uit te dagen. Er ligt ook genoeg goedkope hardware als Arduino en de Raspberry Pi klaar waar steeds meer Q'ers mee willen spelen. Een team bij w00tcamp (ons jaarlijkse hackathon) heeft zelfs een camera gemaakt die een foto maakt als je van onze glijbaan in Amsterdam glijdt. Dus geeking out op dit soort dingen ligt prima in ons straatje.

Auto's met API's en smart charging

Dat besturen van je huis met internet-connected spul wordt overigens soms wel domotica of home automation genoemd. Steeds meer apparatuur komt online. Philips Hue is daar een eerste stap in, maar wat als je koelkast, meterkast, thermometer of auto een connected device wordt? De Tesla Model S, die een eigen dataverbinding heeft, is voorzien van een API waarmee je direct data uit de auto kunt krijgen en zelfs bepaalde functionaliteit van de auto op afstand kunt besturen, zoals de deuren openen of de auto laten toeteren. Kijk maar: <http://docs.timdorr.apiary.io/>

Laatst hebben we een week lang gewerkt aan een prototype voor een smartphone app voor Cohere (<http://cohere.eu>), een startup verbonden aan YES!Delft die zich bezighoudt met alles dat te maken heeft met het laden van elektrische auto's. Door in je meterkast een apparaat te installeren dat direct koppelt op je electriciteitsmeter en die data naar de cloud stuurt, kun je live op je smartphone zien hoeveel energie je verbruikt. Prima, op zich bestaan zulke apps al. Maar Cohere wil meer: met de app kun je beïnvloeden hoeveel van die energie naar je elektrische auto wordt gestuurd. Zo kun je bijvoorbeeld instellen dat je



De Tesla Model S. Toch wel de beste auto ééer.



je auto zo snel mogelijk opgeladen wilt hebben, of dat de auto moet laden op basis van de energie die door je zonnepanelen wordt opgewekt. Smart charging dus. Dankzij die week prototypen heeft Cohere een mooi voorbeeld kunnen laten zien aan investeerders en zijn ze echt geholpen om een pie-in-the-sky idee concreet te maken.

iBeacons

Weet je wat een iBeacon is? Dat is een nieuwe manier van inter-device communicatie, voorgesteld door Apple. Als een apparaat een iBeacon uitzendt, kunnen andere apparaten in de buurt die oppikken, grofweg bepalen hoe ver weg ze zijn, en op basis daarvan een actie uitvoeren. Bijvoorbeeld push notifications: je staat voor een Starbucks en ontvangt een berichtje. "Laat dit zien en ontvang 25% korting op een latté!"

Je kunt je voorstellen dat dit nogal wat mogelijk maakt. iBeacons an sich zijn niet super origineel. Wat het potentieel revolutionair maakt is dat Apple het introduceert, en Apple heeft best wel wat draagvlak wat betreft het introduceren van zulke standaarden, zoals Webkit. En het is open, dus niet gelimiteerd tot alleen hardware van Apple.

Dit gaat best een impact op de markt hebben, denken wij. Daarom starten we samen met H2 Interactive (het bedrijf van Jan David Hanrath, die ook *OneMoreThing.nl* runt) een nieuwe venture: BeaconHQ. We zijn al in gesprek met musea, bibliotheken en supermarkten die wellicht als launching partner kunnen aanhaken. En dan gaan we lekker prototypen. We zijn aan het proberen of indoor positioning te doen is door te trianguleren. Dan zijn wayfinding (indoor route) apps ook ineens te doen!

Web 3.0, real-time en Meteor

Als je tegenwoordig terugkijkt op hoe technologie zich op het internet heeft ontwikkeld, kun je één specifieke trend identificeren die veel impact heeft gehad: de zogenaamde Web 2.0 beweging. Web 2.0 is weliswaar een beladen verzamelnaam die het soms moeilijk maakt om te weten waar het nou precies over gaat. Maar voor ons is het de opkomst van rich web applications dankzij de technische innovatie van XMLHttpRequest, ofwel AJAX. Dat was in 2005 en luidde een nieuw tijdperk van applicaties op het internet in. Gmail en Google Maps waren daar voorlopers in en lieten zien dat apps op het web net zo krachtig als (of misschien wel krachtiger dan) die op desktops kunnen zijn.

Ondertussen is iedereen op zoek naar "Web 3.0". Er was lang een theorie van de "Semantic Web": een internet waarbij data gestructureerd is, zodat allerlei sites en apps makkelijk met elkaar kunnen praten. Web 3.0 is er op die manier niet gekomen. In plaats daarvan hebben we grote sociale netwerken zoals Facebook, waarbinnen alle data semantisch ontsloten wordt via een API. Handig, maar niet gelijk aan de pipe dream van het Semantic Web.

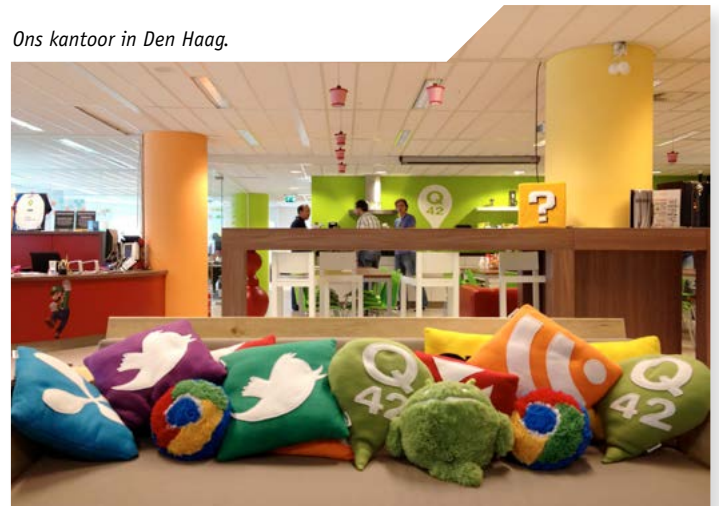
Wat zal dan de volgende grote technische stap zijn online? Wij denken aan real-time apps. De meeste apps laten data zien die afkomstig is van een server, waarbij je de pagina moet refreshen om nieuwe data te tonen. Met de komst van real-time verandert dat en wordt live data van de server naar de client gepusht. Goede voorbeelden van real-time apps? Wat dacht je van Twitter, Gmail of gewoon chat. Zulke apps zijn wel te bouwen met AJAX, maar het blijft lastig omdat er geen échte real-time laag is. Je moet steeds opnieuw aan de server via polling vragen om data. Real-time blijft de uitzondering.

Gelukkig wordt het wel steeds makkelijker door de opkomst van webstandaarden als WebSockets en WebRTC, waarmee low-level afspraken zijn gemaakt over hoe je data live tussen server en client, server en server en zelfs met WebRTC peer-to-peer kunt versturen.

Maar deze standaarden zijn inderdaad nog erg low-level. Daarom is er nog ruimte voor een "jQuery van real-time": een framework of platform dat het bouwen van real-time apps tientallen keren makkelijker en goedkoper maakt. Meteor (<http://meteor.com>) is daar een voorbeeld van. Met Meteor maak je per definitie real-time apps zonder dat je daar over hoeft na te denken. Wij zijn fan: we hebben al een aantal meetups over Meteor bij ons op kantoor georganiseerd en zelfs de Q42 website (<http://q42.nl> - de source staat op Github: <http://github.com/q42/q42.nl>) ernaar omgeschreven. Naast de ervaring die we er zo mee opdoen levert dat nog een paar voordelen, zoals dat de lijst met Q'ers nu real-time kan worden bijgewerkt. Of dat de kleur van de header-balk gesynct is met de kleur van de Philips Hue lampen bij ons op kantoor in Den Haag.

Real-time apps zijn ook aan de server-kant een uitdaging, omdat je moet leren omgaan met potentieel veel gelijktijdige traffic. Gelukkig hebben we daar ondertussen ook veel ervaring mee dankzij het bouwen van sites als *9292.nl* en de Staatsloterij, waar soms erg veel traffic langskomt. Die scaling uitdagingen lossen we vaak op met Google App Engine, en ook daar zit volgens ons de toekomst in: platforms-as-a-service waarbij wij als webbureau niet meer hoeven na te denken over het beheren en onderhouden van servers, maar dat allemaal aan Google overlaten zodat wij ons kunnen focussen op het bouwen van de business logic.

Ons kantoor in Den Haag.



Over Q42

Q42 is een happy place voor nerds. We zijn met z'n vijftigen, in Den Haag en Amsterdam, en werken voor klanten en aan onze eigen producten, games en experimentjes. Als je de 9292 app op je smartphone gebruikt, dan gebruik je onze software. Of als je Staatsloten hebt gekocht op staatsloterij.nl, dan ook.

Heb je zin om eens een kijkje te komen nemen bij ons op kantoor? Dat kan. Mail dan even chris@q42.nl en dan prikken we een datum. Btw, we vinden bijklusstudenten en afstudeerders ook rete-interessant. Niet heel gek, aangezien Herman, Jan-Willem en Alexander - alledrie van de TU - je al voorgingen. Kom je ook?



De Rogers-Ramanujan identiteiten

Roel Tielen

In het jaar 1913 zorgde de Indiër Srinivasa Ramanujan Aiyangar (Ramanujan) voor een hoop opschudding door een lijst met identiteiten te publiceren die hij naar eigen zeggen stuk voor stuk had bewezen. Helaas waren het geen bewijzen zoals we deze tegenwoordig kennen: hij kreeg de identiteiten ingefluisterd in zijn dromen van een Indische god. Voor hem genoeg reden om ervan overtuigd te zijn dat de identiteiten correct waren.

Wiskundigen probeerden een tijd lang de identiteiten van Ramanujan daadwerkelijk te bewijzen (of te weerleggen). Bij een tweetal identiteiten kon echter niet vastgesteld worden of deze correct waren of niet:

Voor $|q| < 1$ geldt:

$$1 + \sum_{n=1}^{\infty} \frac{q^{n^2}}{(1-q)(1-q^2) \cdots (1-q^n)} = \prod_{n=1}^{\infty} \frac{1}{(1-q^{5n-4})(1-q^{5n-1})}.$$

$$1 + \sum_{n=1}^{\infty} \frac{q^{n(n+1)}}{(1-q)(1-q^2) \cdots (1-q^n)} = \prod_{n=1}^{\infty} \frac{1}{(1-q^{5n-3})(1-q^{5n-2})}.$$

Hoewel we in dit artikel niet verder in zullen gaan op convergentie van oneindige sommen en producten, is er natuurlijk wel een vraag die de lezer zich zou moeten stellen: Waarom convergeren deze reeksen voor $|q| < 1$?

In het jaar 1917 vond Ramanujan in een oud tijdschrift een bewijs van de twee identiteiten dat Leonard James Rogers al in 1884 had gevonden. Aangezien Rogers zich al lang niet meer bezighield met dit onderwerp, was de ophef over Ramanujans identiteiten nooit tot hem gekomen. Vanaf het moment van de herontdekking werden de twee identiteiten voortaan de Rogers-Ramanujan identiteiten genoemd.

Nadat de Rogers-Ramanujan identiteiten echt bewezen waren, verschenen er in latere jaren meerdere bewijzen. Zie [1] voor een aantal bewijzen van deze identiteiten.

Later bleek dat de Rogers-Ramanujan identiteiten voorkwamen in zeer uitlopende gebieden zoals onder andere de statistische mechanica en de theorie van partities. We zullen ons voor nu richten op de partities.



Partities

Om het verband tussen de Rogers-Ramanujan identiteiten en partities te achterhalen, zullen we eerst beginnen met de definitie van een partitie:

Definitie (Partitie)

Een partitie van een getal $n \in \mathbb{N}_{\geq 1}$ is een verzameling positieve gehele getallen $\lambda_1, \lambda_2, \dots, \lambda_k$ met $k \in \mathbb{N}_{\geq 1}$ zodanig dat:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k \text{ en } \sum_{i=1}^k \lambda_i = n.$$

We noteren een partitie als volgt: $(\lambda_1, \lambda_2, \dots, \lambda_k)$.

Voorbeeld:

Het getal 7 heeft vijftien verschillende partities, te weten:

$$(7), (6, 1), (5, 2), (5, 1, 1), (4, 3), (4, 2, 1), (4, 1, 1, 1), (3, 3, 1), (3, 2, 2), \\ (3, 2, 1, 1), (3, 1, 1, 1, 1), (2, 2, 2, 1), (2, 2, 1, 1, 1), \\ (2, 1, 1, 1, 1, 1), (1, 1, 1, 1, 1, 1, 1).$$

We kunnen onszelf nu een aantal voor de hand liggende vragen stellen:

- Hoeveel partities heeft een getal N ?
- Hoeveel partities heeft een getal N wanneer we eisen dat de λ_i enkel even mogen zijn? Of enkel oneven?

Het moge duidelijk zijn dat de restrictie dat de λ_i enkel even dan wel oneven mogen zijn makkelijk vervangen kan worden door andere restricties. Ook onder die andere restricties kan het interessant zijn om te weten hoeveel partities een getal N heeft.

We zullen vanaf nu gaan kijken naar twee specifieke restricties. Als eerste gaan we kijken naar de partities waarbij de elementen bij deling door 5 rest 1 of 4 geven. Daarnaast zullen we kijken naar de partities waarbij de elementen minstens twee verschillen.

Even een oefening voor de lezers:

Noteer voor het getal 6 alle mogelijke partities onder de zojuist genoemde restricties. Wat valt u op?

Wiskundig onderlegd als u bent, vertrouw ik erop dat na het beantwoorden van bovenstaande vraag, u voor de getallen 1 tot en met 20 hetzelfde doet. U zult dan zien dat voor ieder getal van 1 tot en met 20 het aantal partities onder beide restricties gelijk is.

Na de vaststelling dat de gelijkheid ook voor het getal 1 tot en met 20 geldt, ligt de volgende vraag voor de hand:

Is voor ieder getal N het aantal partities waarbij de elementen onderling minstens 2 moeten verschillen gelijk aan het aantal partities waarbij de elementen na deling door 5 rest 1 of 4 geven?



Het blijkt inderdaad zo te zijn dat dit voor ieder getal N geldt. Het bewijs hiervan werd gevonden in het jaar 1980 door A. Garsia en S. Milne. Aangezien het een 50 pagina's lang bewijs is, zullen we het hier achterwege laten. Het globale idee is dat er een bijectie wordt geconstrueerd voor alle N tussen de ene en de andere restrictie. Zie ook [2]

De oplettende lezer zal zich na al deze woorden enigszins bedrogen kunnen voelen. Het grootste gedeelte van de tekst ging tot nu toe over partities en een gelijkheid tussen twee soorten van deze partities. Maar wat is nu precies het verband tussen de partities en de Rogers-Ramanujan identiteiten? Om dit verband te kunnen vinden, dienen we ons eerst te concentreren op zogenaamde *genererende functies*.

Genererende functies

Laten we beginnen met een definitie.

Definitie (Genererende functie)

Een genererende functie is een functie die, indien geschreven als machtreeks, aanleiding geeft tot een rij a_n . Hierbij ontstaat de rij a_n door te kijken naar de coëfficiënten van de machtreeks:

$$\sum_{n=0}^{\infty} a_n x^n.$$

Voorbeeld: $\sum_{n=0}^{\infty} (ax)^n$ is een genererende functie voor de rij $1, a, a^2, \dots$

Het voordeel van genererende functies is dat we een oneindig lange rij kunnen vangen in slechts een enkele formule. In het geval van de partities zullen we, gegeven een restrictie x , een rij $\{a_n\}$ op de volgende manier construeren:

$$a_i = (\text{aantal partities van } i \mid \text{restrictie } x)$$

Bij iedere restrictie van de λ_i kunnen we op deze manier een rij $\{a_n\}$ construeren en voor deze rij een genererende functie bepalen. We bekijken eerst de genererende functie van de partities waarbij de elementen onderling minstens twee verschillen. Deze partities worden ook wel *2-verschillend* genoemd. Het bewijs van onderstaande stelling is te vinden in [3].

Stelling

Een genererende functie van het aantal partities die 2-verschillend zijn is

$$g(x) = 1 + \sum_{n=1}^{\infty} \frac{x^{n^2}}{(1-x)(1-x^2) \dots (1-x^n)}.$$

Voor de partities waarbij de elementen na deling door 5 rest 1 of 4 geven leiden we ook een genererende functie af. Hiervoor gebruiken we onderstaand lemma. In verband met een beperking van het aantal woorden staat het bewijs wederom in [3].

Lemma (Aantal partities met elementen uit: $\{a_1, a_2, a_3, \dots\}$)

Een genererende functie f voor het aantal partities van n waarbij de elementen enkel mogen komen uit de verzameling $\{a_1, a_2, a_3, \dots\}$ is de functie

$$f(x) = \frac{1}{\prod_{i=1}^{\infty} (1 - x^{a_i})}.$$

Stelling

De genererende functie van het aantal partities waarbij de elementen gelijk zijn aan $\pm 1 \pmod{5}$ is

$$g(x) = \prod_{n=1}^{\infty} \frac{1}{(1 - x^{5n-4})(1 - x^{5n-1})}.$$

Bew. We maken handig gebruik van het feit dat we de genererende functie kennen van de verzameling $\{a_1, a_2, a_3, \dots\}$. Bekend is dat de elementen gelijk aan $1 \pmod{5}$ van de vorm $5n - 4$ met $n \in \mathbb{N}_{n \geq 1}$ zijn. Ook is bekend dat de elementen gelijk aan $4 \pmod{5}$ van de vorm $5n - 1$ met $n \in \mathbb{N}_{n \geq 1}$ zijn. Hieruit volgt direct dat de genererende functie gelijk is aan:

$$\prod_{n=1}^{\infty} \frac{1}{(1 - x^{5n-4})(1 - x^{5n-1})}.$$

□

We hebben nu van beide soorten partities de genererende functie afgeleid. We weten, uit het bewijs van Garsia en Milne, dat voor iedere N het aantal partities onder beide restricties gelijk is. Hieruit volgt dus ook dat de genererende functies van beide soorten partities gelijk moeten zijn! Op deze manier verkrijgen we onderstaande gelijkheid:

$$1 + \sum_{n=1}^{\infty} \frac{x^{n^2}}{(1-x)(1-x^2) \dots (1-x^n)} = \prod_{n=1}^{\infty} \frac{1}{(1 - x^{5n-4})(1 - x^{5n-1})}$$

We herkennen onmiddellijk de eerste Rogers-Ramanujan identiteit!

Wanneer we de tweede Rogers-Ramanujan identiteit bekijken, zal het u niet verbazen dat ook deze identiteit een gelijkheid is tussen de genererende functies van twee soorten partities. Kan u ook ontdekken welke twee soorten partities een rol spelen bij deze gelijkheid?

Nawoord

Aangezien dit artikel slechts een fractie kan bevatten van een compleet bachelorproject, heb ik een keuze moeten maken welke facetten van de Rogers-Ramanujan onderwerpen ik wel en niet wilde behandelen. Zo zijn de werken van Ramanujan zeker de moeite waard om te lezen. Hij heeft zelf vele boeken geschreven met daarin tal van wonderlijke formules. Verhalen over zijn leven zijn ook erg interessant om te lezen.

Dit bachelorproject is vorig jaar uitgevoerd bij de analyse groep onder begeleiding van Wolter Groenevelt. Mede vanwege de goede begeleiding, waarvoor nogmaals dank, is het uiteindelijk gelukt om dit bachelorproject met succes af te ronden. 

Referenties

- [1] G.E. Andrews, q-series: Their development and application in analysis, number theory, combinatorics, physics, and computer algebra, American Mathematical Society Providens, Rhode Island, 1986
- [2] A. Garsia en S. Milne, A Rogers-Ramanujan bijection, Journal of Combinatorial Theory, Volume 31, 1981, blz. 289-339
- [3] G.E. Andrews en K. Eriksson, Integer Partitions, Cambridge University Press, 2004



De structuur van wilde en tamme verzamelingen

Auke Booij

Wiskundigen maken zich vaak druk om “wilde” verzamelingen die als tegenvoorbeelden worden gebruikt voor bepaalde intuïtie. We kunnen ons beperken tot een kleiner universum met slechts “tamme” verzamelingen, maar dan moeten we ook andere “tamme” verzamelingen uitsluiten. Een hapje Leidse wiskunde over yoghurt met zwarte klontjes.

Ten behoeve van de orthonormaliteit van mijn wiskundige kennisbasis heb ik besloten mijn bacheloreindproject in Leiden te doen. Leidse wiskunde heeft onder Delftenaren het imago nogal abstract en stoffig te zijn, en alhoewel ik niet zal ontkennen dat hier geen kern van waarheid in zit, is hun wiskunde erg leerzaam en bijzonder inspirerend. In dit artikel hoop ik jullie hier iets van te kunnen laten proeven.

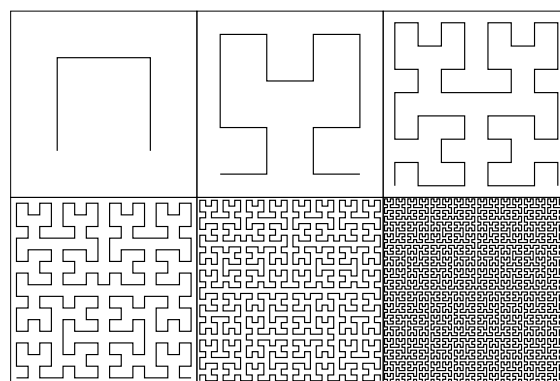
Dit artikel gaat over “wildheid”, in de zin dat er soms contraïntuïtieve situaties optreden in de wiskunde. Wildheid en tamheid zijn hierin geen exact gedefinieerde begrippen, maar een collectie van observaties van raar of niet-fysiek gedrag van wiskundige objecten. Ik ga hier voorbeelden van geven, en dan pogen om dit soort situaties te voorkomen, en eerst lijkt dat te lukken, maar dan toch weer niet, en dan is alweer het einde van het artikel en sluit ik af met één of andere grappige conclusie.

Wilde verzamelingen

Eerst een woordje over verzamelingen voor die enkele verdwaalde ziel die hier nog nooit mee bezig is geweest. Waar ik mij in dit artikel mee bezig houd zijn *deelverzamelingen*, en dit zijn in feite “kleuringen” van objecten. Stel dat je een wit object hebt, zoals een whiteboard, een witte veter of een zwembad volle yoghurt (respectievelijk $\mathbb{R}^2$, $\mathbb{R}$ en $\mathbb{R}^3$), dan is een *deelverzameling* daarvan simpelweg een bepaalde manier om een gedeelte daarvan zwart te verf.

Echter blijkt dat in het wiskundige geval, waarin deze objecten niet uit een eindig aantal atomen maar een oneindig aantal punten bestaat, er enkele rare dingen gebeuren. Zo kan men — wiskundig gezien — een oneindig dunne zwarte draad zo breien dat het hele whiteboard bedekt wordt. Dit is een “wilde” verzameling in de zin dat je nu dus een surjectieve functie van een lijnstuk naar een vlak kunt opschrijven: je maakt van de één-dimensionale draad een vlak, en alhoewel dit het hele punt van breien is als je een draad met dikte hebt is het nogal opvallend dat je dus ook kunt breien met een oneindig dunne draad.

Een bloemkool is opgebouwd uit wit vruchtvlees (of heet dat bloemvlees?), maar ook uit leegte die opgevuld wordt door lucht. Wiskundigen maken zich erg druk om hun ontdekking dat je een knolselderij kunt opdelen in een eindig aantal bloemkolen, en die op zo’n manier weer in elkaar kunt steken zodat je uiteindelijk twee knolselderijen hebt. Oké, opnieuw lukt dit niet in het echt omdat men dan simpelweg een eindig aantal atomen heeft, welke behouden dient te worden, maar (sommige) wiskundigen kunnen wel een bol in $\mathbb{R}^3$ opdelen in een eindig aantal deelverzamelingen, en deze zo verplaatsen dat men uiteindelijk twee bollen overhoudt. Dit is het Banach-Tarski paradox, en is opnieuw zo’n situatie waarin “wilde” verzamelingen optreden.



Figuur 1: Het whiteboardvullende garen van David Hilbert. Dit zijn de eerste zes stappen van het oneindig lange breipatroon.

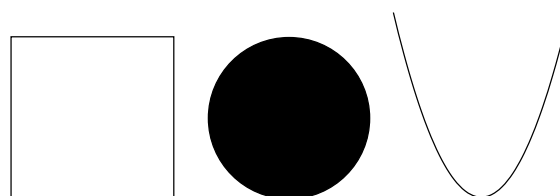


Figuur 2: De verdubbeling van bollen, door Stefan Banach en Alfred Tarski, 1924, papier.

Voor de wiskundigen onder ons met kennis van maattheorie: in het geval van Banach-Tarski zijn sommige van deze deelverzamelingen (dwz. bloemkolen) noodzakelijk onmeetbaar — anders zou de som van de volumes van de deelverzamelingen niet behouden blijven. Sowieso zijn onmeetbare verzamelingen trouwens “wild”.

Tamme verzamelingen

Ik hoop dat als je dit leest, je het met me eens bent dat de voorbeelden hierboven vreemde situaties illustreren, en daarmee “wild” genoemd kunnen worden (opnieuw, dit is dus geen exact begrip maar een collectie van contraïntuïtieve observaties). Vraag is dan wat het andere uiterste is: wat zijn “tamme” verzamelingen?



Figuur 3: Simpele meetkundige deelverzamelingen, te weten: de rand van een vierkant, de cirkelschijf, en een stukje van een parabool

Bovenstaande verzamelingen zijn in veel opzichten erg tam: ze zijn meetbaar (in de zin van maattheorie), eenvoudig op te schrijven (voor wiskundigen



althans), hun constructie gebruikt niet het keuzeaxioma (dit is een wiskundig akkefietje waar veel wildheden gebruik van maken), ze hebben meetkundig eenvoudige eigenschappen, etc.

Nu zijn er natuurlijk veel meer nette, meetkundige verzamelingen, en we kunnen ons afvragen of we hier een duidelijke wiskundige klasse van kunnen maken. We vinden soelaas in de *semialgebraïsche verzamelingen*.

Semialgebraïsche verzamelingen

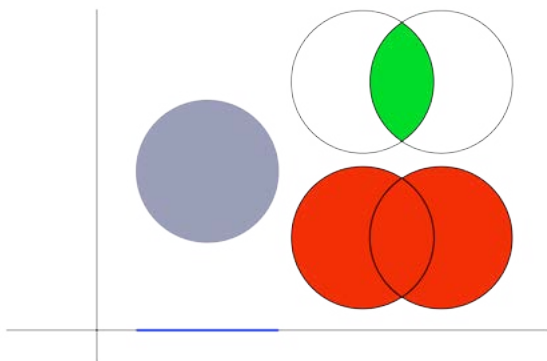
Een *semple semialgebraïsche deelverzameling* van $\mathbf{R}^n$ is één van de vorm:

$$\{x \in \mathbf{R}^n : f(x) = 0, g_0(x) > 0, g_1(x) > 0, \dots, g_k(x) > 0\}$$

waarbij f en de g_i polynomen zijn in de variabelen $x_0, x_1, \dots, x_{n-1}$. Voorbeelden van dit soort verzamelingen zijn de open disk (volgens $\{x \in \mathbf{R}^2 : x_0^2 + x_1^2 < 1\}$), een lijnstuk $(0, 1)$ (volgens $\{x \in \mathbf{R} : x > 0, -x + 1 > 0\}$) en de grafiek van een parabool (volgens $\{x \in \mathbf{R}^2 : x_0^2 - x_1 = 0\}$). Eindige verenigingen van zulke *semple semialgebraïsche deelverzamelingen* noemen we *semialgebraïsche verzamelingen*, en alle meetkundige objecten zijn van die vorm: strandballen, de rechthoekige vlapakken van de Albert Heijn met zo'n handige schroefdop, de kartonnen soepkommen van Sodexo die, zoals iedereen weet, iets groter zijn dan de stenen en dus de voorkeur verdienen, noem het maar op.

Structuren

Het blijkt dat er nu iets interessants geldt: als we twee zulke *semialgebraïsche verzamelingen* nemen, zeg $A \subset \mathbf{R}^n$ en $B \subset \mathbf{R}^n$, dan zijn hun vereniging $A \cup B$, hun doorsnijding $A \cap B$, en het complement $\mathbf{R}^n \setminus A$ ook semialgebraïsch. Met $A \subset \mathbf{R}^n$ en $B \subset \mathbf{R}^m$ geldt ook dat $A \times B$ semialgebraïsch is, en de *projectie* van A , te weten $\pi_n(A) := \{(x_0, \dots, x_{n-2}) : (x_0, \dots, x_{n-1}) \in A\}$, ook.



Figuur 4: Een cirkelschijf (grijsblauw), de doorsnijding van twee schijven (groen), de vereniging (rood), en de projectie van de grijsblauwe schijf (blauw lijnstuk). Ik studeer geen IO.

Het lijkt er dus op dat we ons mooie universum van tamme verzamelingen gevonden hebben, waarin we de gebruikelijke operaties op verzamelingen kunnen toepassen, en bovendien (voor meetkundigen erg belangrijk) kunnen projecteren. Zo'n universum noemen we een *structuur* of *model* (en dit zou bij logici een belletje kunnen doen rinkelen, alhoewel we dezelfde objecten hier in een volledig andere context zien).

Overigens dient het vermelding dat de meeste “interessante eigenschappen” van hierboven vrij eenvoudig te bewijzen zijn. Voor de projectie-eigenschap hebben we echter moeten wachten op een zeer technisch bewijs van de logicus Alfred Tarski, die we ook eerder al bezig zagen aan bloemkool en knolselderij.

Uitbreiden van structuren

Echter zitten veel redelijk “tamme” verzamelingen niet in deze *structuur* van *semialgebraïsche verzamelingen*. Zo is de verzameling van gehele getallen, $\mathbf{Z}$, gezien als deelverzameling van $\mathbf{R}$, geen semialgebraïsche verzameling, terwijl $\mathbf{Z}$ — afhankelijk van wie je het vraagt — niet bijzonder wild is. Wat gebeurt er dus als we $\mathbf{Z}$ toevoegen aan de structuur? Met enkele doorsnijdingen met semialgebraïsche verzamelingen en projecties (voor een exacte constructie zou je mijn scriptie kunnen lezen) volgt dan dat *alle* aftelbare verzamelingen in de structuur zitten. Met andere woorden: door aan de bijzonder tamme structuur van semialgebraïsche verzamelingen simpelweg $\mathbf{Z}$ toe te voegen hebben we een explosie veroorzaakt van allerlei wilde verzamelingen die we dan ook toe moeten laten in ons universum.

We kunnen dus geen structuur opstellen waarin slechts nette algebraïsche en meetkundige verzamelingen zitten. Als we dus de rareiteiten van Hilbert's breipatronen en goocheltrucs met bloemkool willen uitsluiten, zonder meteen een heleboel wiskunde uit het oog te verliezen, kan dat niet op deze manier — een enigszins teleurstellende conclusie van al dit werk, die laat zien hoe dicht onze alledaagse wiskunde verbonden is met rare situaties, en die laat zien dat we ons altijd moeten hoeden voor gebruik van intuïtie in het rekenen. *It is the only way.*

Conclusie

Het is iets na lunchtijd en ik zit met een broodje in de trein van Leiden naar Delft. Mijn begeleider heeft me net wat simpele (juist de simpele!) maar confronterende wiskundige vragen gesteld over mijn BEP. Antwoorden dat je iets “wel ziet” is in Leiden natuurlijk nooit voldoende, en totdat je een bewijs hebt opgeschreven is het wat mijn begeleider betreft niet bewezen en misschien zelfs onzin. Dan heb je aan mij een goede; ik maak überhaupt geen aantekeningen. Ze leven daar van bewijs naar bewijs, en werken aan hun theorie zoals wij werken aan een stuk MATLAB-code: regel voor regel. Je hoeft geen genie te zijn voor hun wiskunde, als je maar de tijd neemt.

Toch kijk ik nog even naar mijn rechterhand: ook nog vanmiddag prof. Hendrik Lenstra een hand gegeven. Onder zekere kringen is hij een bekende man, voornamelijk vanwege de toepassing van zijn LLL-algoritme in analyse van cryptografie. Het respect waarmee je in Leiden betrokken wordt in wat alleen maar beschreven kan worden als “echte wiskunde” vond ik verfrissend, en overall om je heen werken mensen aan de wiskundige kennis van zichzelf en de wereld.

En toch: er knaagt iets. De stof van het onderwerp, mijn begeleider, de omgeving, ik vind het allemaal geweldig, maar *waarom*? Waarom maken we ons zo druk om de wilde verzamelingen en de logica achter contraintuïtie? Nee, de Leidse algebravakken zijn geweldig, en ik raad jullie allemaal van harte aan om vakkken als topologie te volgen (zeker als je (buitenland)ambitie hebt: bij de meeste wiskundeopleidingen is het basiskennis), maar geef mij maar een echte vraag over computers of de Schrödingervergelijking. Mocht er dan iets wilds gebeuren zien we dat dan wel.

Referenties

- [1] Auke Bart Booij, *De structuur van tamme en wilde verzamelingen*, 2013.

Gadgets

Herman Banken

Naast de gave gadgets van Kickstarter schrijf ik in dit nummer ook wat over een zelfgemaakte gadget. Elk jaar in november worden er op w00tcamp van Q42 de gaafste dingen gemaakt. Voor het eerst in de 3.5 jaren dat ik er werk was ik er bij. Wat een energie! Tussen teams die onder andere werkten aan glijbaancamera's, de Oculus Un-Rift en een Meth Cook Simulator maakte ik met mijn team:

PAMela The sexy Presence Awareness Machine

In de avond van vrijdag 8 november op 9 november werd ze geboren: PAMela. PAMela is "the sexy Presence Awareness Machine". Ze weet waar jij bent en is je op diverse manieren van dienst.

Jaap, Arjen, Chris, Laurens, Korjan, Jeroen en ik hebben een Raspberry Pi omgetoverd tot een apparaatje dat je slechts in je LAN hooft te pikken om te kunnen zien wie er aanwezig is. De Pi stuurt MAC-adressen die via WiFi of via broadcast pings worden opgevangen door naar een api in de cloud. Hier wordt bijgehouden welke MAC-adressen er bij een persoon horen.

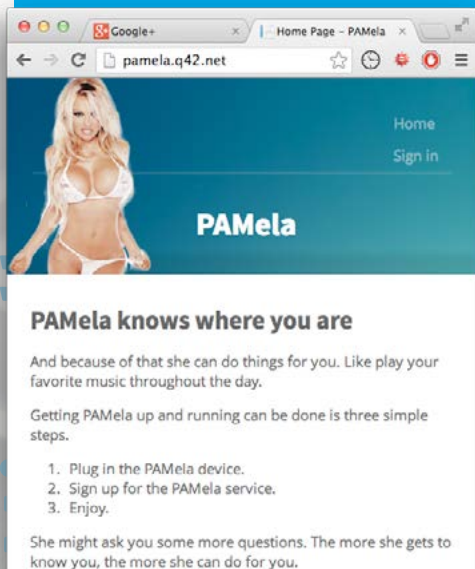
We hebben er voor gezorgd dat PAMela is uit te breiden, met siliconen. Siliconen worden automatisch op elke Pi geïnstalleerd en ontvangen events als er personen aankomen of vertrekken. Wij hebben alvast siliconen gemaakt voor Sonos: speel alleen de favoriete muziek af van de mensen die aanwezig zijn; en Philips Hue lampen: pas de belichting aan aan de smaak van de aanwezigen. Ook hebben we een plugin gemaakt voor de TV die in het kantoor hangt: je kan daar nu op zien wie er in Den Haag is en wie er in Amsterdam aan het werk is.



PAMela is nu nog een Raspberry Pi, maar in de toekomst ziet ze er misschien zo uit.

Het mooie is dat PAMela door de siliconen voor van alles gebruikt kan worden. Onder andere het aansturen van de apparaten in de volgende kolom:

zie pamela.q42.net en w00tcamp.nl



KICKSTARTER

Energie

We leven in een tijd waarin steeds meer wordt nagedacht over energieverbruik. We willen dat de lampen automatisch uitgaan als we weggaan en dat de verwarming het huis net warm heeft als we thuis komen. En dat kan ook. Een notificatie krijgen als je een apparaat aan hebt laten staan? Kan. Zien welk apparaat het minst zuinig is? Kan. Dat het echt hip is dat blijkt wel uit alle Kickstarter projecten de laatste tijd:

Plugg.ee

Gebaseerd op de Spark die ik in een eerder MaCHazine al heb genoemd is Plugg.ee, een slim stopcontact. Binnenin vind je een aangepaste Spark, een 110/220V naar 3.3V converter en een relay. Plugg.ee koppelt op de Spark API en cloud en heeft zelf een app waarmee je apparaten kan aan- en uitzetten op afstand. Met de krachtige ARM Cortex M3 processor kunnen in de toekomst nog vele slimmigheidjes worden toegevoegd. Zoals bijvoorbeeld de melding dat je kookplaat nog aanstaat terwijl jij aan het joggen bent. Bekijk Plugg.ee op <http://kck.st/1aCHkYF>



Neurio

Leuk allemaal, die apparaten per stopcontact, maar je hebt er natuurlijk meerdere van nodig. Enter Neuroio. Eén apparaatje, dat je in de meterkast aansluit, meet welke apparaten er aan staan. Zonder elk afzonderlijk apparaat aan te sluiten op een sensor. Neuroio werkt door het verschil in verbruik voor en na het inschakelen van een apparaat te relateren aan dat bepaalde apparaat. Je lamp op je bureau? Verbruikt 8 watt. De koelkast? 30 watt. Je computer staat te renderen? 1000 watt. Erg slim bedacht! Zie: <http://kck.st/19PaGGT>



Sprav

Heb je de hele dag gezwogen met complexe functieringen of mathematische statistiek, is je database systeem na 5 uur ploeteren nog steeds niet klaar? Dan ben je vast toe aan een lekkere douche. Maar niet zonder dat je bijhoudt hoe lang en hoe warm je doucht! Sprav is een clipje dat je op de slang van de douchekop monteert en met een temperatuur sensor de watertemperatuur meet en met akoestische feedback bepaald hoeveel water er stroomt. En dat allemaal dan ook nog eens met Bluetooth deelt! Zie: <http://kck.st/15MSrk0>



Tracking CHanges

Bastiaan Grisèl

Hands-on kennis die je gelijk in je projecten kunt toepassen, dat is waar dit artikel over gaat. In deze en ook volgende edities van het MaCHazine kun je lezen over tools, methoden en andere technieken dit je kunt gebruiken in je persoonlijke of professionele projecten. In dit artikel is aandacht voor versiebeheer, waarom je het zou moeten gebruiken, wat de mogelijkheden zijn en hoe je ermee aan de slag kunt.

Versiebeheer?

Zeker als informaticus zal je wel eens een aantal hobbyprojectjes hebben gedaan waarbij je op een impulsief moment een gave app of website in elkaar hebt gedraaid. Meestal werkt het allemaal prima en heb je een aan het eind van de rit een in elkaar gehackt stukje software dat doet wat je wilde. Maar dan net op het punt dat je bezig bent die ene feature toe te voegen, gaat ineens alles stuk. De stukjes code die je aan het begin hebt geschreven geven ineens rare errors en het geheel stort als een kaartenhuis ineen. Was er maar een mogelijkheid om een spreekwoordelijke startknop in te drukken en de optie “restart from last checkpoint” te kiezen. Was er maar een soort control-z knop waarmee je terug kon naar het punt dat je stukje software nog werkte, om zo langzaam de nieuwe functionaliteit toe te voegen. Een version control system (VCS) stelt je in staat om dat op een eenvoudige en elegante manier te doen.

De achterliggende gedachte

Wat je met een VCS kan doen, is een soort momentopnames (commits, in VCS-taal) maken van een bepaalde map op je computer. Dit kan bijvoorbeeld de hoofdmap zijn van een project waar je mee aan het werk bent. Je moet zelf aangeven wanneer je een commit maakt, meestal is dat nadat je een stukje functionaliteit hebt geïmplementeerd. Doordat een VCS tekstuele bestanden kan analyseren en source files eigenlijk altijd tekstueel zijn opgebouwd, houdt een VCS bij wat er tussen de commits is veranderd in bestanden. Het mooie van een VCS is dat je vervolgens op elk moment terug kan gaan naar hoe de map eruit zag bij een bepaalde commit.

Versiebeheer is niet alleen mooi als je in je eentje aan een project werkt, juist ook als er meerdere mensen bij een project betrokken zijn is een VCS enorm handig. Bijna iedereen kent de “conflicted copies” op Dropbox als twee mensen tegelijkertijd een documentje aanpassen. Stel je voor dat je dat met je programmeercode zou hebben, dan zou steeds maar één iemand aan een bestand kunnen werken. Gelukkig biedt een VCS verschillende mogelijkheden om bestanden samen te voegen en kan dit vaak zelfs automatisch.

Net zoals bij Dropbox werkt een VCS vaak met een centrale opslagplaats (remote repository) waar een kopie van je lokale map (local repository) wordt opgeslagen. Eens in de zoveel tijd synchroniseer je je lokale map met die van de centrale opslagplaats en dan worden de wijzigingen die jij hebt gedaan centraal doorgevoerd en de wijzigingen die anderen hebben gedaan bij jou doorgevoerd.

Verschillende vormen

Er zijn veel verschillende versiebeheersystemen met elk hun eigen visie en functionaliteit. In de basis doen ze allemaal ongeveer hetzelfde, het grote verschil zit vaak in de rol van de centrale opslagplaats. Bij conventionele gecentraliseerde versiebeheersystemen is er een duidelijk onderscheid tussen de local repositories en de remote repository. Alle informatie over welke commits er zijn geweest en welke bestanden daar zijn aangepast staat opgeslagen op de remote repository. Ook als je een commit doet, dan wordt deze opgeslagen op de remote repository en er gebeurt er verder niets speciaals met je lokale map.

Met gedistribueerde versiebeheersystemen (DVCS) is er geen verschil tussen de local en remote repositories. Dit heeft er mee te maken dat ook elke local repository alle informatie over de commits bevat, waar deze informatie bij een gecentraliseerd versiebeheersysteem alleen op de remote repository bestaat. Om toch een centrale opslagplaats te hebben waar iedereen mee synchroniseert, wordt er meestal één local repository, die ergens op een computer staat die altijd aan het internet is verbonden, als main repository aangewezen. Het voordeel van een DVCS is dat je niet je project kwijt bent, mocht er iets met de computer gebeuren waar de remote repository op draait, iedereen heeft immers een lokale kopie met alle informatie.

Getting started

De bovenstaande alinea's geven een korte introductie van de basisconcepten van een VCS. Als je zelf eens wilt kijken hoe je een zelf een VCS kan gebruiken, neem dan even een half uurtje de tijd en lees het hoofdstuk ‘Aan de slag’ van de bijbel van het meest gebruikte DVCS, genaamd Git op <http://git-scm.com/book/nl>. Als je dit hebt gelezen en je hebt Git geïnstalleerd, probeer het dan eens toe te passen bij een hobbyprojectje en misschien later bij een project van je studie; je zult er enorm veel van leren. Dit is precies zo'n tool die niet alleen tijdens je studie enorm nuttig is, maar ook je professionele carrière een kickstart kan geven. 



Alumnus

Ir. Kees Zeeman

Ik ben in 1983 begonnen met Informatica in Delft toen het nog Technische Hogeschool heette. Informatica bestond net een paar jaar en kwam voort uit Wiskunde.

Na wat onderhuur her en der in de eerste paar maanden heb ik nog een paar maanden in de winter doorgebracht in een caravan bij het Sportcentrum. Ik was dan blij als het weer ochtend was: lekker opwarmen in de college zalen. In februari werd ik ingestemd op de oudraadtweg 31. Dat was een verdieping van 15 mensen die geen lid waren van studieverenigingen. Daar heb ik een hele leuk tijd gehad en nu nog gaan we met zes huisgenoten jaarlijks een aantal dagen oud en nieuw vieren met onze gezinnen. Ook voetbal ik nog steeds bij het sportcentrum bij Ariston80.

Het was de tijd van de eerste IBM PC's met MS-DOS. Je betaalde toen nog de hoofdprijs voor een PC die je halve bureau in beslag nam. Ik kocht mijn eerste PC tweedehands voor drieduizend gulden in 1985. Daarvoor kreeg ik de volgende configuratie: monochroom, harddisk van 20MB en een floppy disk van 360KB met het achtergrondgeluid van een stofzuiger. Netwerken ging via modems over de telefoonlijn (net als bij de fax).

In '89 studeerde ik af bij Prof. dr. I.S. Herschberg op het gebied van computerveiligheid. Hij stond bekend als de "krakende professor". Hij had leuke opdrachten vanuit de industrie voor ons om de veiligheid van systemen te testen. Ik heb samen met een huisgenoot een encryptie-beveiliging voor harddisks gekraakt. Mijn afstuderen deed ik bij TNO Defensieonderzoek op het gebied host-authenticatie (computers die elkaars identiteit vaststellen, voordat ze een sessie tussen hen toestaan). Later is de door mij ontwikkelde software nog toegepast bij de Engelse banken voor transactieverkeer. Wat me nog goed bijstaat is de ondoorgrondelijkheid voor mij van sommige algoritmes

om berekeningen te versnellen. Ik kon met de beste wil van de wereld niet bevatten wat ze nou deden en hoe, maar ze werkten perfect! Diep respect voor de wiskundigen onder ons!

Als ik toen had kunnen vermoeden dat internet zo exponentieel zou groeien en zo'n impact zou hebben op veiligheid en privacy dan was ik daar zeker in verder gegaan, maar toen zag ik daar weinig toekomst in. Herschberg voorzag dat wel en de risico's voor onze privacy, nu ik dit schrijf vind ik dat best opmerkelijk!

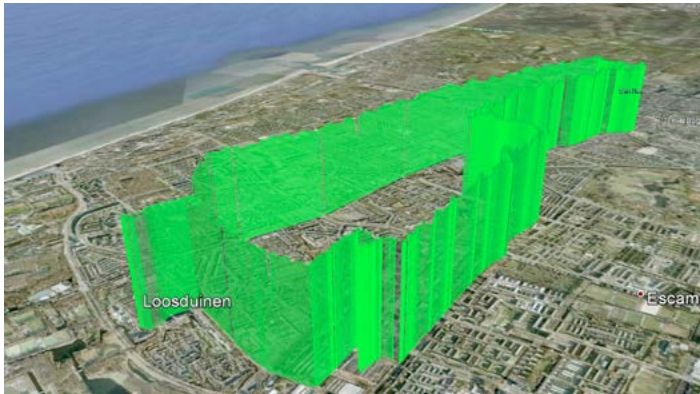
Na mijn afstuderen heb ik nog drie jaar bij TNO gewerkt in de Divisie operations research. Daar ontwikkelde ik simulatie systemen voor onderzoek naar wapen-effectiviteit van luchtverdedigingssystemen, onderhoudsconcepten van F16's en rijkswegen. Toen de oorlog tegen Irak uitbrak (de eerste Golfoorlog) hebben we nog zitten rekenen aan de beste plaats voor de Nederlandse luchtdoelraketten in Turkije, zodat ze goede Line-of-Sight hadden. Ook keek ik toen veel naar CNN, vooral om te zien hoe de patriot systemen in Israël het deden tegen de SCUD-raketten van Irak. Mijn simulaties gaven hetzelfde beeld: de patriots raakten pas recht boven hun eigen lanceerplatform de SCUDs, terwijl ze doelen op 80km kunnen detecteren. Het klinkt raar, maar vanuit je vak kijk je toch ook met een andere bril (valideren van modellen) naar oorlogen dan als burger.

Na TNO heb ik als consultant zes jaar gewerkt voor CMG. Het was de tijd dat detacheerders als paddenstoelen uit de grond schoten. ICT-kennis was schaars, dus de tarieven goed. Het mooie van die tijd was dat je daardoor als consultant ook als een soort alleskunner werd gezien. Ik kon kiezen uit een hele trist van rollen, projecten en opdrachtgevers als adviseur, proces verbeteraar, projecteider of interim manager. Het waren lange dagen met veel reistijd. Toen mijn dochter werd geboren kreeg ik daar genoeg van: ik zag haar door de weeks vrijwel niet.

In die periode werd ik benaderd door Cisco Systems voor een sales-functie. Dat was een zeer inspirerende tijd met ruimte voor goede balans tussen werk en prive, omdat ze 100% stuurden op resultaat en niet op aanwezigheid. Al in 1999 voorzag de CEO John Chambers dat *"Internet is going to change the way we work, learn, live and play"*. Dat is op een onvoorstelbare manier uitgekomen. E-learning kan nu op alle niveau's, de gaming industrie en social media zijn in beurswaarde ge-explodeerd. En op het trieste af (zelfs tijdens etentjes of feestjes), zijn mensen on-line. Cisco groeide mee op de uitrol van internet met routers, switches, IP-telefonie, video-conferencing en allerlei systemen om verkeer op internet efficiënt af te handelen.

Zelf heb ik bijgedragen aan migraties van X.25 naar IP en consolidatie van datacenters van het PolitieNet in Nederland, en van CCTV en telefonie over IP, onder andere voor de gemeente Rotterdam, het havenbedrijf R'dam en Europol. Voor Europol was de reden voor IP-telefonie veiligheid: ze konden dan spraken met dezelfde apparatuur encrypten als het dataverkeer.

Na zes jaar Cisco was ik toe aan een nieuwe uitdaging. Ik wilde eens kijken hoever ik het kon brengen als ondernemer. Omdat organisaties allemaal vrij ver waren met interne automatisering en informatisering, zagen we een gat in de markt op informatiedeling in ketens. Samen met een compagnon heb ik innovaties in publieke en private ketens rond veiligheid geïnitieerd als architect en technisch projectleider, zoals evenementveiligheid bij Sail in 2010 en Parkpop in 2012, transportveiligheid op luchthavens en in tunnels en bij gevaarlijke stoffen tijdens containertransport over water.



Figuur 1: Visualisatie spanning op bovenleidingnet

Sinds 2010 heb ik een tweede bedrijf opgezet waar we als technology provider systemen voor energy, health en safety ontwikkelen. De reden voor mij was om na te gaan of ik als ingenieur in staat zou zijn om met een eigen bedrijf met van de grond af aan zelf ontwikkelde producten in voor mij nieuwe markten succesvol te kunnen zijn. Inmiddels hebben we meerdere fabrieken, trams en gebouwen operationeel op energy en health monitoring en volgend jaar gaan we ook operationeel in de safety hoek met alarmeringssysteem tussen Tata Steel en de Veiligheidsregio Kennemerland. Dat doen we samen met Thales en CityGIS. Waar ik de komende jaren een interessante markt zie voor ons is om energie-prestaties te gebruiken als voorspeller voor onderhoud (z.g. condition-based maintenance binnen assetmanagement). Als studenten of alumni daar ook kansen in zien, neem dan aub contact met me op (kees.zeeman@gmail.com)!

Ik kom nu in aanraking met alle aspecten van EWI in onze producten en diensten: big-data, networking-protocollen, sensing, analyse, micro-controllers, I/O-controllers embedded software, antenne technologie, filters, voedingsconcepten, etc. Wat dat betreft sluit de samensmelting van Electrotechniek, Wiskunde en Informatica zeer goed aan op ontwikkelingen in de markt. De belangrijkste les die ik heb geleerd is dat het heel veel tijd en energie kost om ICT-systemen zo robuust te maken dat ze jaren blijven werken zonder noemenswaardig beheer of onderhoud.

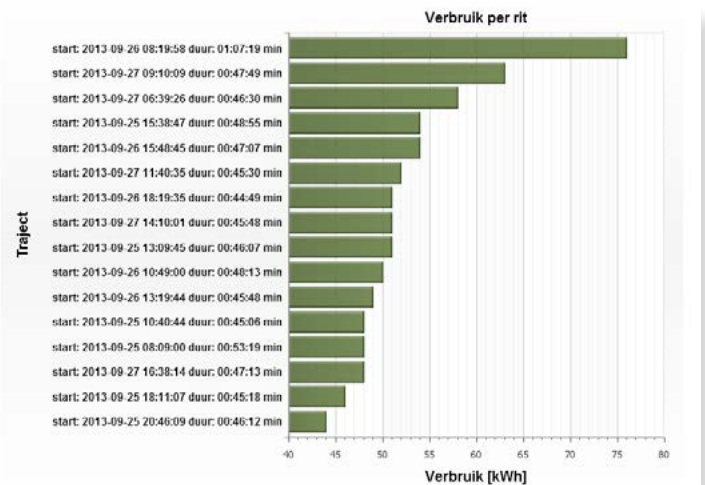


Figuur 2: Embedded programmeren

Mijn rol in het bedrijf is toepassingen bedenken en uitwerken voor bepaalde markten en de bijpassende partners daarvoor te vinden voor ontwikkeling en verkoop, maar ik help ook mee bij installaties en testen.

Wat ik in mijn hele loopbaan heb ervaren is dat je als ingenieur een zeer brede basis hebt, waarmee je snel in staat bent om hoofd en bijzaken van elkaar te scheiden, onderscheid te maken tussen oorzaak en gevolg (geloof me daar hebben hordes mensen moeite mee..) en oog te hebben voor belangrijke details, want daar kan je behoorlijk over struikelen.

Tijdens mijn studie had ik eigenlijk geen goed idee wat ik concreet zou kunnen doen als ingenieur. Nu is me duidelijk dat ingenieurs echt overal nuttig werk kunnen doen! 



Figuur3: Tramverbruik per rit



Figuur 4: Data-acquisitie in tram

Historical Figure: Qin Jiushao

Cathaline Meloen



This article concerns a truly brilliant mathematician who seems to have enjoyed all facets of life. Known for his disrespect to authority, his aggressive nature as well as for his many love affairs and his skills in writing poetry, his affinity with other forms of art as well as sports, Qin Jiushao is a mathematician worth mentioning.

Area

Between 960 and 1279, China was led by the Song Dynasty, under whom's guidance Chinese innovation in science and technology triumphed. During the govern of the Song dynasty, there was a strong military force nationwide (three quarters of the states revenue was spend on military expenses). This resulted in a great need for technical and mathematical creativity and innovation, and government officials would be selected rather on their abilities and talents than on ancastry. Qin Jiushao was born in 1202. His father fulfilled several functions for the government, and Qin moved with him to different states during his youth and early adulthood.

Education

In the preface of his most famous work *Shushu Jiuzhang* Qin Jiushao wrote: "In my youth I was living in the capital, so that I was able to study in the Board of Astronomy; subsequently, I was instructed in mathematics by a recluse scholar." Except for this quote there is little known about his education. The name of his mathematics teacher remains unknown, however we do know that he read the *Nine Chapters on the Mathematical Art*. This book is worth mentioning, since it has played a major role in the development of Chinese mathematics. It is a very practical book that contains 246 mathematic problems intended to provide solutions for day-to-day problems. It covers the subjects engineering, surveying, trade and taxation.



Wealth

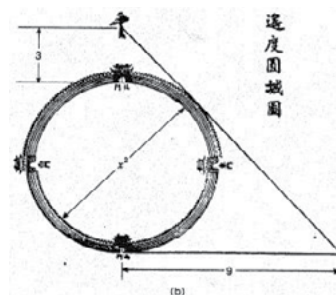
In 1219, when Qin was 17 years old, he volunteered for the army that was currently putting down a rebellion. His many talents and aggressive nature made him successful within the army, and he became a commander defender during his military service. After leaving the military, Qin works as a sheriff, an administrator in Qizhou, as a governor of Hui-chou, and after that many other governmental functions succeeded one another. The rapid changes of function and environment where caused by Qin's abuse of power, his criminal activities (like poisoning his enemies), that led to resistance in the areas where he worked. As a result of his criminal activities and despite his frequent relocations, Qin managed to become very wealthy.

Mathematical Treatise in Nine Sections

In September 1244 Qin's mother dies and Qin left his post in Nanking for the mourning period. During this mourning period Qin wrote *Shushu Jiuzhang*, his *Mathematical Treatise in Nine Sections*. The similarity with the title "Nine Chapters on the Mathematical Art" is not coincidental, however Qin's writing is far more sophisticated. Each chapter describes nine problems. Especially the first four chapters contain extraordinary methods that exceed by far the mathematical knowledge until then.

In its first chapter "Indeterminate equations", Qin presents a general form and complete proof for the Chinese Remainder Problem. In chapter two "Heaven phenomena", he wrote the earliest explanation that describes how Chinese calendar experts calculated astronomical data based on the timing of the winter solstice (lowest point of the sun). In "Area of land and field" Qin presents his "Qin Jiushao's formula" that is similar to Heron's formula, that enables one to calculate the size of the surface of a triangle with the length of the sides.

Chapter four (Surveying) contains the following problem: "Given a circular walled city of unknown diameter with four gates, one at each of the four cardinal points. A tree lays three li north of the northern gate. If one turns and walks eastwards for nine li immediately on leaving the southern gate, the tree just comes into view. Find the circumference and the diameter of the city wall.



Qin obtains the equation: $X^{10} + 15x^8 + 72x^6 - 864x^4 - 11664x^2 - 34992 = 0$
Solution: $x = 3$, so diameter of city is $x^2 = 9$ li " [8]

It is almost impossible to imagine that someone living a life full of turmoil somehow had the peace of mind to write a book in which he solves problems using tenth degree equations and so many other revolutionary proofs and theories (some of which were rediscovered only centuries after his writing). For those who want to read more, check out the references. 

References:

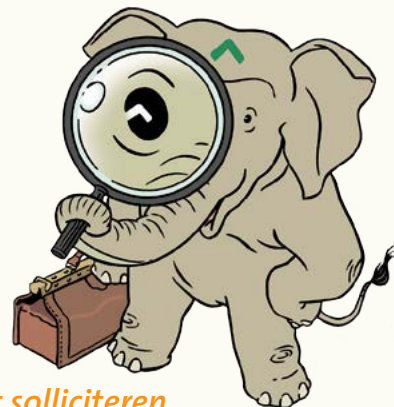
- [1] http://www-history.mcs.st-andrews.ac.uk/Biographies/Qin_Jiushao.html
- [2] http://www-history.mcs.st-andrews.ac.uk/HistTopics/Nine_chapters.html
- [3] http://www.astro.rug.nl/~nevenzeel/Study/PGvdW_t=9C_HCM_a=RB,KN.pdf
- [4] http://en.wikipedia.org/wiki/Mathematical_Treatise_in_Nine_Sections
- [5] http://en.wikipedia.org/wiki/Heron's_formula
- [6] <http://www.maa.org/publications/periodicals/convergence/using-problems-from-the-history-of-mathematics-problems-representing-transitions-in-mathematical>
- [7] http://en.wikipedia.org/wiki/Chinese_Remainder_Theorem
- [8] http://www-history.mcs.st-andrews.ac.uk/Biographies/Qin_Jiushao.html

Bestuur 57 wenst
jullie prettige
kerstdagen en een
gelukkig 2014!



Waarom DSW?

Zorgverzekeraar én IT-leverancier



Doe de 'Beste Werkgever-Test' en weet binnen vijf minuten of je moet solliciteren.

Is jouw ideale werkgever een toegankelijke organisatie waar je trots op kunt zijn?

☐ Ja ☐ Nee

DSW telt ongeveer 450.000 verzekerden, een kleine verzekeraar ten opzichte van de vier grote partijen. Juist door onze kleinschalige organisatie zijn wij in staat de menselijke maat te hanteren. En toch zijn we groot genoeg om belangrijk te zijn in zorgverzekeraarsland. Samen zetten we onze producten en service neer: solidair, klantvriendelijk en anti-bureaucratisch. Daar zijn onze medewerkers trots op!

Houd je van persoonlijk contact met collega's?

☐ Ja ☐ Nee

Bij DSW heerst een familiecultuur, iedereen staat gelijk aan elkaar. We hebben wel ruim zeshonderd collega's, maar iedereen kent elkaar. We hebben uiteraard een directie en ook de afdelingshoofden en teamleiders zorgen voor een goede verbinding in het bedrijf. Maar iedereen zit in één pand en er heerst een open deur-cultuur: als je iets wilt bespreken, kun je ook bij een afdelingshoofd gewoon naar binnen lopen.

Deel je graag je eigen werktijden in?

☐ Ja ☐ Nee

Bij DSW doen we niet moeilijk als je een keer een uurtje eerder weg moet of later wilt beginnen. Je bent flexibel in je werktijden, dat uurtje mag je gerust een andere keer inhalen. Andersom kun je overuren compenseren voor een vrije dag.

Werk je graag met nieuwe technieken?

☐ Ja ☐ Nee

Bij DSW werken slimme mensen die het leuk vinden om nieuwe dingen te proberen. Dat geldt ook voor de IT-afdeling. Nieuwe technieken worden snel getest en toegepast bij geschiktheid. Dit doen we allemaal zelf. Windows applicaties bouwen we met .net en wpf. Web applicaties met .net mvc, knockout en html5. Qua responsive design lopen we voorop. De servicegeoriënteerde architectuur houdt ons flexibel, maar ook het feit dat we niet afhankelijk zijn van externe leveranciers.

Heb jij je eigen stijl en ga je liever niet in apenpak naar je werk?

☐ Ja ☐ Nee

Bij ons hoef je niet in driedelig pak te verschijnen, maar is elke dag casual friday.

Wil je graag je eigen ideeën kunnen uitvoeren?

☐ Ja ☐ Nee

Als jij een goed idee hebt, krijg je van DSW al snel de vrijheid om dit daadwerkelijk uit te voeren. Goede ideeën worden niet alleen door de directie bedacht, maar juist door onze medewerkers. Jij als bedenker kunt dit idee het beste uitvoeren, samen met je collega's. Van DSW krijg je hier alle steun in.

Houd je van ruimte om te experimenteren?

☐ Ja ☐ Nee

Nieuwe ontwikkelingen houden je creatief en leergierig. Natuurlijk kan het dan gebeuren dat er dan een foutje wordt gemaakt. Van belang is hoe er met fouten wordt omgegaan: Als eerste moet je samenwerken om het probleem op te lossen. Daarna moet je de oorzaak vinden. Niet om de schuldige aan te kunnen wijzen, maar om er van te kunnen leren. Zo blijven we allemaal scherp.

Wil je keuzevrijheid in je arbeidsvoorwaarden?

☐ Ja ☐ Nee

Naast een dertiende maand, vakantietoeslag en reiskostenvergoeding, kun je bij DSW zelf kiezen welke voorwaarden je wilt. Zo kun je via DSW een fitnessabonnement of een fiets aanschaffen of kun je een toeslag inruilen voor extra vrije dagen.

Ben je leergierig en wil je doorgroeien?

☐ Ja ☐ Nee

Bij DSW vinden we de kwaliteit en het vermogen van onze medewerkers van het grootste belang. Daar investeren we in. Daarom krijg je alle mogelijkheid voor opleidingen, in zowel geld als tijd. Soms krijg je er bij jouw functie automatisch een opleiding bij. Eigen verzoeken voor studies worden soms wel tot honderd procent vergoed.

Daarom DSW!

Als je (bijna) alle vragen met 'Ja' hebt kunnen beantwoorden, dan is DSW jouw ideale werkgever en willen wij graag kennis met je maken! Vraag om meer informatie, kijk op www.werkenbijdswnl of solliciteer meteen op een van onze vacatures! Heb je de meeste vragen met 'Nee' beantwoord, dan kun je beter bij een groot, bureaucratisch, internationaal bedrijf solliciteren.

*There's no place like
127.0.0.1,
maar 83.98.228.4
komt dichtbij!*

DSW
zorgverzekeraar

goed voor je