

Windows Vista ligt eindelijk in de winkel. Het lijkt een goed moment om terug te kijken naar de manier waarop Windows Vista tot stand gekomen is. Software Release Magazine sprak met Sven Hallauer, Release Manager en Director van het LifeCycle Team over een onmogelijk softwareproject: het werken met 7.500 ontwikkelaars aan de ontwikkeling van een nieuwe versie van Windows.

interview

Windows Vista: the making of

Vierendertig miljoen basic code blocks

Sven Hallauer, oorspronkelijk afkomstig uit Duitsland, kwam via Zuid-Afrika, na tussenstops in Dubai en London uiteindelijk in Redmond terecht. Hallauer's carrière liep dan ook via een minder gebruikelijke weg. Hij volgde zijn vader in 1992 naar Zuid-Afrika en begon daar biologie, wiskunde en chemie te studeren. Hij verveelde zich echter en ging als supportmedewerker werken bij een net geopende vestiging van Microsoft aldaar. Oorspronkelijk wilde hij dat maar voor anderhalf jaar doen, maar het beviel goed en spoedig reisde hij voor Microsoft de halve wereld af. In Dubai deed hij consulting voor grote infrastructuurprojecten én ontmoette hij zijn huidige vrouw. Na een tijd in Londen werd hij naar Redmond verplaatst.

KWALITEITSDOELEN

Hallauer: 'In Redmond was ik .NET global evangelist voor global accounts. Heb dus in 2001 toen .NET begon met grote accounts als Shell, Siemens, Volkswagen Coca Cola gewerkt. Daar startte ik .NET projecten, en droeg ik de technologie uit. Mijn volgende carrièrestap was development manager voor een project dat Microsoft Systems Architecture heette, in principe *prescripted guidance* voor enterprise klanten: hoe je datacentra opbouwt, en hoe je die met Microsoft en third party oplossingen managed, deployt, met Cisco en anderen. Daarna ben ik naar Windows gegaan en heb tweeënhalft jaar geleden het Longhorn-project overgenomen (*Longhorn was de codenaam voor Vista, red.*) en sindsdien was ik verantwoordelijk voor het project- en releasema-

nagement voor Windows Vista. De job is relatief moeilijk te beschrijven, maar het is in ieder geval projectplanning: ik zet alles vast, de release criteria, vergewis me ervan dat ze gehaald worden, bepaal de *schedules* voor de public releases, werk samen met de teams aan de kwaliteitsdoelen voor de individuele features en ben ook verantwoordelijk voor de *sign off* van het project, het bij wijze van spreken op de knop drukken en zeggen: zo, nu wordt het gereleased.'

Het meest opvallende onderdeel dat uit de feature set van Vista verdween, was WinFS. Dat is niet de eerste keer want ik kan me nog herinneren dat het al voor Windows Cairo was aangekondigd (Cairo was een groot Microsoft project tussen 1991 en 1996 waarvan delen in Win95, NT 3.5 en 4 terecht zouden komen, red.).

Hallauer: 'WinFS is nog steeds de droom waar we naartoe willen, maar de technologie is zo complex, dat het niet past binnen de tijdlijnen waarin men een product wil uitbrengen. Het *Longhorn*-project heeft twee verschillende aanlopen gekend. Op de PDC in 2003 hebben we WinFS en andere technologieën gepresenteerd. Wanneer ik daar indertijd verantwoordelijk voor geweest zou zijn, dan zou ik het waarschijnlijk anders hebben gebracht, want veel aanwezigen dachten dat het bijna klaar was. In werkelijkheid werden hoofdzakelijk incubatieprojecten gepresenteerd: alle innovatieve ideeën die we hadden en die we ooit zouden willen *shippen*. De bezoekers dachten echter dat het allemaal in Vista zou komen. De andere dingen zijn voor het grootste



Foto: Microsoft

Sven Hallauer, Release Manager en Director van het LifeCycle Team voor Vista

gedeelte ook terecht gekomen in Vista: Avalon, managed code integratie, presentation platform, maar WinFS heeft het niet gered, dat is eenvoudigweg niet uit de incubatiefase gekomen.'

Waarom niet? Zou het een te ingrijpende verandering met zich mee hebben gebracht, omdat het file systeem aan de basis van alles staat, of zat het te diep in het besturingssysteem?

Hallauer: 'We hebben heel veel dingen diep binnenin het besturingssysteem veranderd. Daarbij gaat het om de balans: hoe ontwikkel je het platform verder, terwijl je aan de andere kant genoeg opwindende eindgebruikersfeatures ontwikkelt, zodat eindgebruikers zeggen:

"Hé, ik wil het hebben vanwege dat feature". Dat we een heel goed deployment en servicing verhaal hebben, is interessant voor enterprise-klienten en voor de OEM's, maar voor de consumenten maakt het niet veel verschil. Of we nu update.exe gebruiken of die XML-files om Windows-updates te downloaden, dat interesseert de eindgebruiker niet veel.'

Hij heeft er wel wat aan maar hij weet het niet.

Hallauer: 'Ja, hij heeft er wel wat aan, maar het is ook beter wanneer hij niet ontdekt hoe gecompliceerd alles is en waarom het nu beter is. Want dat moet op de achtergrond gebeuren. Dat WinFS niet in de tijdsplanning zou passen, dat werd geloof ik al in augustus 2004 duidelijk. Ik weet ook niet wat de huidige planning voor WinFS is, met dat project heb ik niets te maken, dat is weer helemaal naar de incubatiefase, er is een team dat eraan werkt maar wanneer het komt weet ik niet. (Volgens een Microsoft-blogger zou het in Katmai komen, de volgende versie van SQL Server, volgens anderen is dat project een stille dood gestorven, red.)

ROOD POTLOOD *Wanneer we naar de planning van Vista kijken, vanaf het opstellen van de eerste requirements, hoe is dat gegaan, ook methodologisch?*

Hallauer: 'De eerste ideeën, de features werden ontwikkeld nog vóórdat ik ermee van doen had. Toen we het project opnieuw bekeken in augustus 2004 hebben we gezocht naar pijlers waarop we de release willen opbouwen, die we echt binnen twee jaar af kunnen hebben. Toen hebben we het rode potlood genomen en hebben dingen geschrapt. De overblijvende dingen leverden in onze optiek genoeg op om een goede release in goede kwaliteit voor elkaar te krijgen.

Toen hebben we gezegd: wanneer we eind 2006 het product klaar willen hebben, hoe plannen we dat? Wanneer moeten we bèta 1 shippen, hoeveel procent van de features willen we afhebben in bèta 1, hoeveel in bèta 2, hoeveel release kandidaten hebben we nodig? We hebben het engineering proces veranderd, hebben *quality gates* ingevoerd, we hebben een nieuw versioning systeem opgezet vergeleken met wat we bij Windows 2000 en XP gebruikt hebben. Met Windows 2000 en XP hadden we drie *build labs* die allemaal op *main* samengevoegd werden, voor respectievelijk client, server en core. Met Vista werd dat naar zeven uitgebreid om een betere verdeling te krijgen, onder meer omdat de hoeveelheid ontwikkelaars enorm gegroeid was.

Hoeveel ontwikkelaars hebben aan Vista gewerkt?

Hallauer: 'In totaal hebben 7.500 mensen aan het project meegewerkt. Een gedeelte daarvan heeft er echter niet de gehele tijd aan gewerkt. Als constant gemiddelde waren het er ongeveer ongeveer 5000.'

Hoeveel regels code bevatte het project?

Hallauer: 'Het aantal regels code weet ik niet, maar het waren ongeveer 34 miljoen basic code blocks.'

Als je naar dat aantal kijkt, dan zou volgens de statistiek de waarschijnlijkheid dat Vista werkt 0 moeten zijn.

Hallauer: 'Als we dezelfde processen gebruikt zouden hebben die we voor Windows 2000 gebruikten, dan zou het ook niet gelukt zijn. Er zijn meerdere processen: we hebben statische code-analysetools gebuikt, pre fixed pre parsed, die eenvoudig geschreven code analyseren en fouten ontdekken, automatisch, en weer naar de ontwikkelaar terugsturen. We hebben quality gates gebruikt die de teams ertoe brengen zowel feature- als stresstests, tegelijk met de features leveren, we dus de test case-ontwikkeling geaccelereerd tegelijk met de code-ontwikkeling, en dat heeft heel veel opgeleverd. Bij vorige versies hebben we hoofdzakelijk pas getest nadat de code klaar was, dus pas later testcases geschreven. Daardoor was er altijd een groot tijdsverschil. We hebben dat geaccelereerd en hebben gezegd: jullie moeten samenwerken en pas wanneer beide (*testers en developers, red.*) klaar zijn, kunnen jullie inchecken. We hebben de *basics* geïntroduceerd, de 16 à 18 *basic area's* van het product, die zaken eenvoudig over het product heen behandelen, zoals security, client performance, server performance, client applicatie compatibility, server applicatie compatibility en IPv6 compliance. Dat heeft zaken geïnstitutionaliseerd, omdat er nu mandaten waren, ieder product moest kloppen. We hebben ook veel nieuws gedaan op het gebied van het in lagen onderbrengen van de architectuur. We hebben de dingen van de grond af aan bekeken en gezegd: dat zijn de architectuurlagen die we nu in Windows hebben. We hebben Server 2003 als basis genomen, omdat dat het nieuwste was. Anders heb je steeds het backporten van bugfixes en dat is gewoon heel veel werk. Bovendien was het iets stabielier én we hebben op die code base voor de eerste keer 64 bit XP geleverd. We hadden immers een 64 bit client basis nodig. We hebben dus naar die architectuurlagen gekeken en hebben gezegd: hoe willen we Windows in de volgende tien tot vijftien jaar ontwikkelen om het qua architectuur beter op te bouwen, grotere stabiliteit te bereiken, meer performancewinst te krijgen en het meer modulair te laten zijn. Aan de ene kant moeten de feature teams gewoon verder kunnen gaan en die gebieden verbeteren, en aan de andere kant mogen nieuwe features die we ontwikkelen het product niet slechter maken (erop achteruit laten gaan). Ze moeten allemaal in die nieuwe architectonische basis ingebouwd kunnen worden.'

ESCROW-FASE Hallauer: 'Al die veranderingen hebben ertoe bijgedragen dat we wezenlijk meer veranderingen konden aanbrengen tijdens de verschillende

Escrow-perioden van het product. Want gedurende de *main line development* heb je *bug bars*, dat zijn de fixes die we willen doen, vervolgens heb je open development features en alle bugs, dan heb je open bugs, en tenslotte beperk je dat alles wanneer je de *milestone* wilt sluiten – bijvoorbeeld bèta 1 of bèta 2 – en dan ga je de Escrow-fase in. Tot nu toe hebben we gedurende een Escrow-periode maar heel weinig veranderingen doorgevoerd, omdat we het systeem gewoon niet goed genoeg begrepen om de precieze gevolgen van het veranderen van een binary te kunnen zien. We hebben in de laatste paar jaar eenvoudig tools ontwikkeld die ons precies vertellen wanneer we die verandering in de binary aanbengen: wat zijn de totale *code parks* (gebieden) die daardoor beïnvloed worden. Daardoor konden

'Alle veranderingen in het engineeringproces hebben ons echt vooruit geholpen'

we gedurende Escrow meer veranderingen doorvoeren en het product veel stabielier maken (wezenlijk) als het anders mogelijk geweest zou zijn. We wisten precies: die change heeft impact op die drie testteams. Dan konden we gericht naar de testteams toegaan en zeggen: hé, dit team heeft deze change gemaakt, jullie moeten nu die code park nog eens natesten om er zeker van te kunnen zijn dat die niets kapot gemaakt hebben wat daarvoor gefunctioneerd heeft. Al die veranderingen in het engineeringproces (ontwikkel/productie/ontwerpproces) hebben ons echt vooruit geholpen. Het heeft ervoor gezorgd dat we toch alles onder controle gebracht hebben, hoewel het product zelf zo gegroeid is. Het media-center is er nu aan toegevoegd, de tablet pc functies. Alle producten die voorheen separaat geleverd werden, zijn nu allemaal in de Windows Vista codebase geïntegreerd. De feature set is gegroeid én we hebben alle producten die we na XP uitgeleverd hebben in Vista geïntegreerd. SP2, de firewall, mediacenter, de nieuwe browser, tablet pc features, om dat allemaal samen te brengen krijgen, was al mooi.'

Dus het is gebouwd op basis van Server 2003 en niet op XP?

Hallauer: 'XP leek tamelijk sterk op Windows 2000, daar zat ook maar anderhalf jaar tussen. We hebben heel veel gedaan op het gebied van de modules. We hadden voor het eerst een feature inventory tool in Longhorn/Vista, voor iedere feature apart. Dat hielp ons ook met de andere opgave die we hebben. We hadden een enorme hoeveelheid *penetration testers* op Vista, die gekeken hebben naar waar hackers kunnen binnendrin-

gen. Het feit dat we alle feature specs en design specs in één afzonderlijke (enkele) databank hadden, heeft enorm geholpen. De *penetration testers* konden die eenvoudig doorlezen en zeggen: "Ha, nu weet ik hoe je geprobeerd hebt het te gebruiken." Ik weet uit mijn andere ervaringen hoe een hacker daarover zou denken. Daardoor hebben we veel bereikt om het geheel veiliger te kunnen maken.'

ERFENIS *Een niet onbelangrijk deel van de zwakke plekken komt nog voort uit de zeer lange geschiedenis van Windows, de erfenis uit het verleden. Hoe is daarmee omgegaan?*

Hallauer: 'Er is geen binary die we niet gewijzigd hebben. Misschien afgezien van third party drivers of zo, ik weet niet of de HP LaserjetII driver veranderd is, maar die krijgen we sowieso van third party's. Maar iedere afzonderlijke binary die Microsoft produceert, hebben we veranderd. Aan sommige zit je ook vast, zoals shell32.dll. die heb je eenvoudigweg nodig voor de miljoenen applicaties erop gebouwd zijn. Alle api's en

'Voor een ontwikkelaar is het heel moeilijk om te zeggen hoe lang het zal duren om een feature te schrijven'

functie calls moeten allemaal bewaard blijven. Aan de andere kant willen we ook innoveren omdat we het nieuwe presentatieplatform hebben, en mensen moeten dus op die oude shell32 opbouwen.'

Zijn die modules nu werkelijk onafhankelijk van elkaar, kun je ze eruit halen en gewoon vervangen door een andere?

Hallauer: 'Niet allemaal, deels wel. We hebben de Internet Explorer browser en de Shell gedeeld, maar het is verder per binary verschillend. We hebben daar gedeeld, bijvoorbeeld bij de browser en de Shell, waar architectonisch gezien de laagopdeling zinvol was. Wanneer de modules conceptueel echt op verschillende niveaus zijn, dan is het zinvol te delen en anders niet. Dat is de hoofdgedachte geweest. Internet Explorer is gewoon een applicatie, we moeten door de compliance met de EU andere browsers kunnen aanbieden die net zo functioneren, we moeten de gebruiker de mogelijkheid geven Internet Explorer te verwijderen. Dus heeft het in de eerste plaats regulatorisch zin, maar het is ook een applicatie. Net zoals bij iedere andere applicatie zou je in staat moeten zijn, in ieder geval theoretisch, ze uit het product te verwijderen. Of het commercieel of qua EU-regelgeving zinvol is, dat is een andere vraag, daar heb ik niet veel mee te maken.'

Hij verdwijnt ook niet werkelijk, maar Neelie denkt dat waarschijnlijk.

Hallauer: 'Ja, de dll's zijn er allemaal nog. Alleen de complete user interface verdwijnt en de hyperlinks.'

Er is dus ook een set regulatorische requirements, daar sta je niet zo bij stil.

Hallauer: 'Bij Windows XP hadden we helemaal geen regulatorische requirements. Voor de DOJ-settlement bestonden geen echte government regulations voor Windows, die ons beperkten in wat we wilden maken. Nu zijn we voor de eerste keer post-DOJ, post-EU, zover dat we van tevoren weten waar we modulair moeten worden omdat we weten dat bepaalde regeringen het eindproduct willen beïnvloeden. De Koreanen bijvoorbeeld hebben ook weer hun eigen eisen. De nieuwe technologieën die we gebouwd hebben, vooral de deployment tools, maken het ons eenvoudig verschillende versies te maken. Wanneer de binary's zo geconstrueerd zijn, kun je op eenvoudige wijze bijvoorbeeld de Media Player of de browser niet ter beschikking te stellen aan de gebruiker. Het helpt ook grote bedrijven die een Windows willen installeren wat qua mogelijkheden beperkt is. Je kunt je voorstellen dat men iemand bij de receptie een versie zonder games wil aanbieden. Maar je kunt ook de omgekeerde weg bewandelen en kiezen voor zo weinig mogelijk functionaliteit. Daar helpen de nieuwe deployment-tools, de modularisering van het besturingssysteem en het gebruik van XML als descriptortaal voor de installatievoorkeuren ons om Windows naar de volgende generatie te brengen.'

OPTIMISTISCH *Ik begrijp dat Microsoft zelf verschillen maakt in de verschillende Windows-vormen en wat daarin opgenomen is, maar hoe ver gaat dat voor bedrijfsmatige eindgebruikers?*

Hallauer: 'Sommige dingen doen we zelf, zoals het feit dat games standaard niet geïnstalleerd zijn bij de business en de enterprise-versie. Dan zijn er zaken als DVD-playback, die niet automatisch bij die versies worden geleverd, maar klanten die beschikken over een enterprise agreement of software assurance kunnen het zogenaamde Kodak-pack toevoegen of via derden op Internet kopen. In sommige bedrijven moet de salesforce hun producten via DVD demonstreren. In de enterprise versie zijn de tablet pc-functies geïnstalleerd, in de business-versie niet. Bij de installatie gaat het om de vraag: wat wil ik in mijn image hebben en wat wil ik via group policy installeren? Dat laatste heeft ook heel veel mogelijkheden dingen te forceren, desktop-achtergronden, screensavers, password detention policy's, en veel meer.'

Meestal zijn ontwikkelaars heel optimistisch en denken dat alles heel snel gaat. Zijn er ondanks dat dingen die sneller gingen dan gedacht?

Hallauer: 'Ja, ontwikkelaars zijn altijd heel optimis-

tisch. Voor de gewone ontwikkelaar is het heel moeilijk om te zeggen hoe lang het zal duren om een feature te schrijven en dat dan ook in goede kwaliteit te produceren. Het andere probleem is dat sommige features afhankelijk zijn van andere. Wanneer ze opgebouwd zijn op andere delen van het systeem die ook veranderd worden, dan wordt het al snel heel gecompliceerd. De mediacenter functionaliteit in Vista is een goed voorbeeld. In bèta 1 weet ik niet eens of het überhaupt gewerkt heeft, in bèta 2 heeft het een beetje gewerkt, maar met veel crashes en *glitches*. In RC1 was het nog steeds niet al te geweldig en dan met RC2 was het in een zeer goede staat. Dat kwam doordat het mediacenter team gebouwd heeft op het nieuwe grafische platform, het nieuwe presentatieplatform én het nieuwe audio-platform. Dat betekent dat alles waarop het gebaseerd was constant veranderde. De onderdelen hadden pas goede kwaliteit in RC1. Dat team had dus de gehele RC2 cyclus nodig om op dezelfde kwaliteit te komen en te zeggen: nu is onze applicatie echt stabiel. Hun features waren wel klaar omdat de verschillende bouwstenen waarop ze gebouwd hebben nog steeds veranderden. Device driver kwaliteit komt ook altijd pas later in de cyclus van een nieuw besturingssysteem aan de orde. De third party's moeten natuurlijk stuurprogramma's schrijven voor de producten die ze nu verkopen, en daarom zullen ze niet hun gehele ontwikkelingsafdeling erop instellen dat Microsoft eventueel binnen een of twee jaar een nieuw product uitbrengt. Natuurlijk, Intel, ATI, Nvidia hebben altijd ontwikkelaars bij ons gestationeerd, maar dat zijn er altijd maar een paar gedurende de product cyclus, zodat zij weten wat er in het besturingssysteem gebeurt, wat de veranderingen zijn die ze op lange termijn moeten doorvoeren. De echte inzet van de ontwikkelcapaciteit vindt echter niet plaats voor er een release candidate is. Afhankelijk van hoe groot die firma's zijn. ATI, Intel en Nvidia zijn zeer groot en hebben grote ontwikkelafdelingen die kunnen vroeger beginnen vergeleken met firma's die kleiner zijn, die pas echt kunnen beginnen met de RTM. Voor hen is het te duur om voor een bèta iets te ontwikkelen dat dan misschien nog veranderd moet worden. Daarom hebben we ook die tijdsspanne tussen RC (8 november) en de global availability launch op 30 januari. Die tijdsspanne geeft ons de tijd om het gehele ecosysteem op te krikken om genoeg device drivers te hebben zodat het ook een echt goede *experience* wordt.'

En die weten ook wat ze moeten doen?

Hallauer: 'Ja de third party's zeker, die weten intussen welke bugs ze hebben, we helpen ze die te prioriteren, van welke bugs wij willen dat ze het eerste opgelost worden. We ontvangen immers de data van alle bèta's en RC's die we gemaakt hebben, zodat we kunnen zeggen waar de crashes en hangs zijn. Wanneer ik naar die

data kijk dan is het overgrote deel nog steeds *third party driver* code. Naar mate de third party's meer gebruik maken van de nieuwe driver-modellen, wordt dat steeds minder en steeds minder ernstig. Dat geeft zoveel ruimte om te ontwikkelen. Met het nieuwe display-driver-model wat we hebben, de WDDM, is het verschil met XP erg duidelijk. Wanneer de driver crasht flikkert het beeldscherm even, misschien gaat Aero weg, maar Windows draait nog steeds. Alles is er nog, terwijl je bij XP een bluescreen krijgt wanneer de driver crasht. Dat is de engineering die we in de modellen gestopt hebben om Windows stabiel te maken tegenover third party drivers, maar wanneer de driver een bug heeft, is die er nog steeds. Het is weliswaar niet meer zo erg voor de eindgebruiker, maar we krijgen nog steeds de data terug en kunnen zeggen: jullie hebben zo en zoveel crashes in die module en dit gebeurt er in het systeem, jullie moeten dat verbeteren. Het delen van de data tussen ons en de third party's is wezenlijk beter in Vista, zodat wij het ecosysteem veel beter kunnen maken. Niemand heeft er uiteindelijk iets aan, wanneer een driver of een ander deel van het ecosysteem niet goed werkt. Dan is de klant er niet gelukkig mee.'

OPT OUT *Het automatisch terugsturen van data bij crashes en zo, heeft dat veel voordeel opgeleverd?*

Hallauer: 'Ja dat heeft waanzinnig veel opgeleverd. Voor ons was het enorm belangrijk. Gedurende de bèta-cycli waren de pc's automatisch in opt in modus, zodat we dat allemaal terugkregen. In RC1 hebben we het veranderd in *opt out* zodat de klant toestemming moet

'Alle producten die voorheen separaat geleverd werden, zijn nu allemaal in de Windows Vista codebase geïntegreerd'

geven om de data te versturen, maar deze datastroom en vooral de grotere bètaprogramma's bij Vista hebben ons enorm geholpen. Voor bèta 2 alleen hadden we meer dan twee miljoen geïnstalleerde bèta2-pc's. Voor RC2 hadden we die pc's plus nog eens een half miljoen. Die bandbreedte van hardware en software die mensen op hun pc's zetten en gebruiken geeft ons de basisdata waarop we moeten letten. Daardoor is Vista qua betrouwbaarheid ook wezenlijk beter dan XP SP2.'

Het lijkt me ook ideaal, want op die manier krijg je alle data die nodig zijn.

Hallauer: 'Ja zolang we die geïnstrumenteerd hebben. Met Windows XP hadden we 20.000 mensen in

het tech bèta programma en die moesten we een cd mailen en zeggen: hier is de laatste milestone. In de eerste plaats kost dat ons meer geld en ten tweede meer tijd. Het duurt immers één tot twee weken tot dat de cd's geproduceerd, verstuurd en geïnstalleerd zijn. We hadden toen ook niet zoveel feedback-mogelijkheden als we nu hebben, we hadden geen SQIM, alleen Watson. Bij Vista hadden we 30.000 mensen in het tech bèta programma en we hebben met opzet altijd geprobeerd steeds op vrijdag de release te hebben. Veel tech bèta enthousiasten kunnen niet wachten om het product in handen te hebben. Omdat ze het niet uit hoofde van het beroep doen maar als

'We hadden veel *penetration testers* op Vista, die gekeken hebben naar waar hackers kunnen binnendringen'

hobby, is vrijdagavond de juiste tijd. Dan downloaden ze het, en wordt het zaterdag op een cd gezet en geïnstalleerd. We hebben installaties gezien waarbij na een release op vrijdag op maandag of dinsdag 50 tot 60 % van de nieuwe build geïnstalleerd was. De snelheid van de feedback en de vele datapunten heeft ons enorm geholpen in het release management.'

Partijen als McAfee en Symantec beklagen zich nu omdat ze geen toegang meer hebben tot de kernel van Vista. In hoeverre is Microsoft van plan ze tegemoet te komen?

Hallauer: 'In de eerste plaats moet ik zeggen dat het daarbij alleen om 64 bit gaat. Ik zou het ook niet de toegang tot de kernel *afsluiten* willen noemen, maar de kernel *beveiligen*. In feite heeft niemand er iets aan dat derden de kernel van Windows veranderen. Ze kunnen het weliswaar goed bedoelen, maar wij kunnen niet meer garanderen dat het product veilig is. Wanneer die een veiligheidsgat hebben en ze hebben directe toegang tot de kernel, dan kan door hun veiligheidsgat het totale product misbruikt worden en dat is niet wat we willen. Daarom willen we de kernel beveiligen. McAfee en Symantec hebben zeker goede bedoelingen, maar wat wanneer het niet McAfee is maar een pappa en mamma-winkel in China en ik misleid word en ik installeer het verkeerde product? Daarom willen we dat voor de volgende versie veranderen. Voor de 32 bit-versie heeft het geen zin. Er bestaat zoveel prehistorische software die dat steeds gedaan heeft. Maar wanneer je twee of drie releases naar de toekomst kijkt en zegt: hé, we zullen ooit in een echte 64 bit wereld zijn, waarschijnlijk niet de volgende versie maar die daarna,

dan moet je nu de juiste strategische keuzes maken voordat jan en alleman 64 bit drivers en applicaties gaan schrijven. In zo'n geval weet je niet wie de kernel aanpast en daartegen willen we onze klanten beschermen. Bovendien geloven we eenvoudigweg niet dat we hun mogelijkheden beperken. Bij 32 bit blijft alles bij het oude en dat zal toch verreweg het grootste gedeelte zijn. Voor mij is het meer een storm in een glas water.'

ACHTERDEUR *Vanuit de optiek van Microsoft is het begrijpelijk, maar naar buiten werkt het anders. Het is ook een politiek probleem. Welke vrijheid geef ik aan de third party's en uiteindelijk aan de klant?*

Hallauer: 'Ja, we hebben nu geprobeerd te zeggen: deze api's kun je wel gebruiken maar dit gedeelte is niet beschikbaar. Maar we hebben, ook met XP SP2, nog steeds het probleem dat de virusscanners en systeemtools vernieuwd moeten worden. Die hebben namelijk allemaal gebruik gemaakt van niet gedocumenteerde API's. Wij wilden nooit dat die extern gebruikt werden, en mensen hebben dat desondanks gedaan. Nu beperken we dat en zeggen: hier zijn de API's die jullie kunnen gebruiken, laat het weten wanneer er nog zaken ontbreken en we passen ze aan. Maar ze moeten het via het officiële kanaal regelen, niet via een achterdeur of via andere delen van het besturingssysteem. Het is goed die vragen te stellen en af en toe gebeurt het dat we iets doen wat niet echt tot het einde doordacht is, maar hier wel dege-lijk.'

Dré de Man