

Personele aspecten zijn belangrijker dan technische aspecten

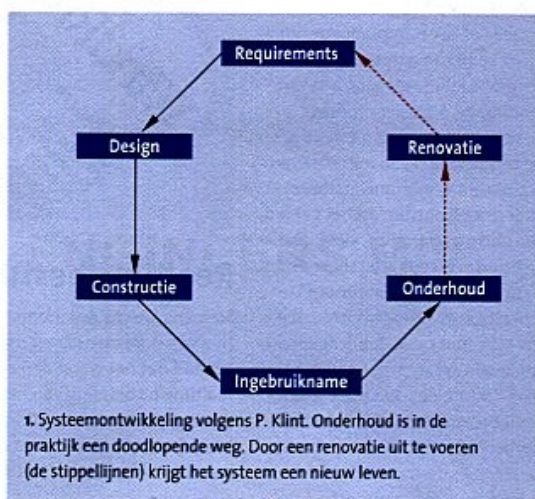
# Software-onderhoud wordt

Het onderhoud van software is belangrijk en zal alleen nog maar belangrijker worden. Toch staat de gemiddelde ontwikkelaar niet te trappelen voor een rol in software-onderhoud. Die is liever met nieuwe dingen bezig. Het is dan ook zaak de vicieuze cirkel van lage status en ongunstige perspectieven te doorbreken.

HANS VAN THIEL

**A**ndermans code aanpassen is nu niet bepaald het ideaal van de software-engineer. En voor wie het nog niet wist heeft prof. dr. P. Klint van het Centrum voor Wiskunde en Informatica (CWI) ook nog eens slecht nieuws: in het jaar 2020 zal nog maar een derde van alle programmeurs aan nieuwe software werken. Nu al houden vier van de zeven programmeurs zich bezig met onderhoud en reparatie van bestaande systemen. Door embedded engineers zal overigens de software-soep wel niet zo heet hoeven te worden gegeten. De genoemde cijfers gelden voor alle computersystemen, inclusief mainframes, en gigantische legacy-problemen komen in de embedded wereld niet of nauwelijks voor.

Dat neemt niet weg dat software-onderhoud ook hier steeds belangrijker wordt. Reden voor het Mikrocentrum er een themadag over te organiseren, met als sprekers: prof. dr. P. Klint van het CWI te Amsterdam, ing. J. Jacobs van Philips Technical Application Software Services uit Eindhoven, R. Ponder van de ICT groep in Son, mr. L. Ritzema van Holland van Gijzen Advocaten en Notarissen te Eindhoven, N. van Mullekom van Origin TA IPS te Veldhoven, ir. A. van Oijen van Océ Technologies in Venlo en tot slot J. Kleijne van VDO



Car Communication in Eindhoven. Deze bedrijven zullen bij embedded engineers waarschijnlijk bekend zijn, behalve dan Holland van Gijzen Advocaten en Notarissen. Onderhoud van software heeft namelijk ook juridische kanten, met name op het gebied van aansprakelijkheid. Hoewel software geen zaak is in de zin der wet maakt embedded software juridisch gezien vaak wel deel uit van een zaak en valt daarmee onder de productaansprakelijkheid. Mr. Ritzema adviseerde om in

**'In 2020 werkt nog maar een derde van alle programmeurs aan nieuwe software.'**

elk geval goede contracten overeen te komen en voor complexe systemen een Service Level Agreement af te spreken.

## Soorten onderhoud

Ook uitbesteden van onderhoud heeft natuurlijk juridische aspecten. Zo doet

Origin het onderhoud van de ASML-software voor wafer-steppers/scanners. Die beslaat ongeveer 5 miljoen regels code en in de loop der tijd is de kennis daarover bij ASML afgenomen en geleidelijk overgegaan naar Origin. Dit heeft er weer toe geleid dat de twee partners inderdaad hun onderlinge verhouding duidelijker willen en zijn gaan vastleggen.

In technische zin bestaan er, volgens J. Jacobs van Philips, drie soorten onderhoud:

- *correctief onderhoud*, kan noodzakelijk zijn door defecten in requirements, design of implementatie;
- *adaptief onderhoud*, het gevolg van gebreken in requirements of design;
- *perfectief onderhoud*, het verbeteren van software die niet echt slecht is (gold-plating of wel vergulding).

Met name voor consumentenproducten is het herstel van fouten in embedded software als het product al op de markt is gekomen, veel en veel duurder dan in een eerder stadium. Zo berekende Jacobs dat een fout die in een half uur door een programmeur gecorrigeerd kan worden (kosten fl. 50,-) dui-



# steeds belangrijker

## Onderhoud is vaak brandjes blussen

zend keer zo veel kost als diezelfde fout tot in de geleverde producten is doorgedrongen. Het loont dus duidelijk de moeite foutpropagaties te voorkomen. En dat kan alleen door een duidelijke kwaliteitsmodellering in te voeren en die vooral ook te implementeren en vast te houden. Het probleem in de praktijk van alledag is dat de tijd het

**‘Langdurige betrokkenheid van de engineers is alleen al door de lange inleertijden een harde noodzaak.’**

vaak wint van de kwaliteit. Jacobs typeerde dit anekdotisch als de Bommelaanse oplossing: ‘Maak snelheid, jonge vriend, de schicht heeft een schrijpend tekort aan brandstof.’

### Discipline en motivatie

Om een kwaliteitssysteem te handhaven - goede ervaringen werden bijvoorbeeld gemeld met software-inspecties naar Tom Gilb en Dorothy Graham - is een strenge organisatorische discipline vereist. Als één ding duidelijk werd op deze themadag over software-onderhoud was het wel dat *organisatorische* en dan vooral *persoonlijke* aspecten belangrijker zijn dan de technische.

Klont schetste bijvoorbeeld een *software development life cycle* (SDLC) waarin onderhoud geen dood spoor behoeft te zijn (zie figuur 1) maar hij meldde dat in de praktijk - weer de praktijk - onderhoud vooral wordt toegewezen aan *junior engineers* of aan de uitgerangeerde collega's. Software-engineers willen graag aan nieuwe ont-

Bij Océ Technologies werken wereldwijd 21.000 mensen die gezamenlijk een omzet genereren van bijna 3 miljard Euro's. Het bedrijf ontwikkelt print-, scan- en kopieerproducten voor professionele gebruikers. Embedded software, variërend van realtime control, beeldverwerking, communicatie-modulen tot gebruikersinterfaces, speelt een belangrijke rol in deze producten.

De organisatiestructuur van Océ kent functionele afdelingen en projecten. Elk project is in principe van tijdelijke aard en omvat, naar behoefte van het project, medewerkers van verschillende afdelingen. Deze vorm van organisatie heet bij Océ *matrixstructuur*.

De traditionele kopieer-apparaten van Océ veranderden niet veel als ze eenmaal in de markt waren gezet. Alleen als (hardware) onderdelen verouderd raakten moesten specificaties wel eens worden aangepast. Bij open systemen als printers en combinaties van printers, scanners en copiers zit de functionaliteit meer in de embedded software. Daardoor moeten er thans regelmatig *bug fixes* en functieuitbreidingen worden geïmplementeerd in systemen die al in bedrijf zijn. Het karakter van onderhoud verschoof van hardware naar software en van reactief naar pro-actief.

Er is bij Océ goed nagedacht over de manier hoe onderhoud dan het best is te organiseren. Als een product is uitontwikkeld stopt het desbetreffende project en worden de medewerkers elders ingezet. Het lag dus voor de hand een aparte onderhoudsafdeling op te zetten en die te financieren door service-contracten. Men zag echter de volgende nadelen hiervan:

- weinig flexibel voor software-engineers en onaantrekkelijk uit carrière-oogpunt;
- lastige kennisoverdracht vanuit bestaande functionele afdelingen, hoewel die wel meer aandacht zouden gaan besteden aan onderhoudsaspecten (een voordeel);
- hoeveelheid werk sterk afhankelijk van kwaliteit van producten en van markteisen,

kans dat de afdeling een soort parallelle ontwikkelafdeling zou worden;

- een langere weg tussen signalen uit de markt en de lopende projecten voor nieuwe producten.

Men heeft daarom als experiment het onderhoud van *wide format printing systems* (WFPS) niet in een aparte afdeling maar in een apart project ondergebracht. Dit 'Maintenance and Sustenance' (M&S) project, dat overigens wel een onbepaalde levensduur heeft, omvat het (software)onderhoud van een dertigtal verschillende WFPS-producten. Drie procent van alle WFPS software-engineers werkt thans binnen M&S. Om software-onderhoud voor hen aantrekkelijk te maken en te houden besteedt het bedrijf aandacht aan de volgende punten:

- laat het belang zien van onderhoud, de huidige producten zijn de financiële basis van het bedrijf;
- bied een technische uitdaging, wat met name kan in de functionele aanpassingen (sustenance);
- bied de M&S engineers een goed carrièrepad;
- benadruk de snelle resultaten die je kunt zien bij onderhoud;
- bevorder de contacten met klanten;
- pas het werk aan aan het persoonlijkheidsprofiel van de engineer.

Onderhoud is vaak brandjes blussen en dat heeft soms een negatieve invloed op medewerkers. Anderzijds bevordert succes en waardering de motivatie. Het management moet ook de sociale aspecten - zoals een feestje als een mijlpaal is bereikt - niet vergeten. De eerste resultaten van het experiment bij Océ geven aan dat de communicatie met de service-afdeling is verbeterd en dat de respons tijden voor problemen zijn verkort.

Bron: A. van Oijen, *Embedded Software Maintenance in Document Processing Systems*, Océ Technologies.

wikkelingen werken en het is moeilijk hen te motiveren voor onderhoud aan bestaande systemen. Klint noemde het, ook gezien het toenemende belang van onderhoud, noodzakelijk de vicieuze cirkel van lage status en ongunstige perspectieven te doorbreken. Bij Océ (zie kader) maar ook bij Origin wordt het probleem erkend en probeert men er dan ook iets aan te doen. Bij Origin, dat zoals gezegd onderhoud uitvoert voor ASML, is een langdurige betrokkenheid van de engineers alleen al door de lange inleertijden een harde noodzaak.

Hoewel *legacy*-problemen, in de zin van bergen ongedocumenteerde Cobol-code, in embedded software niet voorkomen is het toch ook niet zo dat dergelijke systemen maar twee of drie jaar meegaan. Zowel ICT als VDO Car moet rekening houden met onderhoudstermijnen van meer dan tien jaar.

Zo heeft men bij ICT in 1995 een systeem geproduceerd voor de opslag van medische beelden op cd. Het project is

gestart in 1994, een jaar eerder, en dat eerste systeem maakte gebruik van een OS-9, werd ontwikkeld volgens Yourdon, Hatley en Pirbhai, was geschreven in C en werkte met cd-i en mpeg- en jpeg-coderingen. In 1999 is dat systeem gemigreerd naar Windows NT4, ontwikkeld met Rational Rose (OO en UML), geschreven in C++ met standaard-componenten en (voor de kenners) SQA, Purify, Quantify en SODA.

Onverlet de veranderde software en hardware moeten de medische beelden die vanaf 1995 zijn opgeslagen natuurlijk toegankelijk blijven. ICT heeft dan ook met de afnemers ondersteuning afgesproken tot 12 jaar na de EOL (end-of-life) van deze producten. Onderhoud is bij ICT dan ook georganiseerd in een *life cycle management* (LCM) structuur.

Ook de auto-navigatiesystemen van VDO moeten jaren lang worden ondersteund. Hier speelde bijvoorbeeld het geval dat ook oude geïnstalleerde systemen de nieuwe, veel uitgebrei-

dere kaarten en plattegronden moeten kunnen lezen. Autofabrieken hebben zelfs contractueel laten vastleg-

**‘Software-engineers willen graag aan nieuwe ontwikkelingen werken en het is moeilijk hen te motiveren voor onderhoud aan bestaande systemen.’**

gen dat de navigatie systemen tot 7 jaar nadat de productie is gestopt nog worden onderhouden. Ook bij VDO is software-onderhoud dan ook een volwaardige functie binnen de productlevenscyclus. □