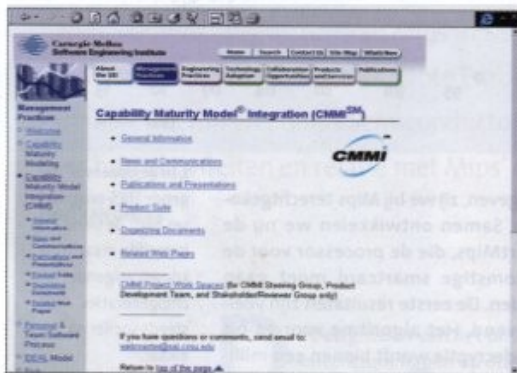


Vooraf in software-industrie groeit belangstelling

Software is much like anything else



Het belang van productontwikkeling neemt toe; het is de kernactiviteit van elke ontwikkelaar van embedded systemen. Steeds vaker heeft productontwikkeling een interdisciplinair karakter. Bovendien worden producten steeds gecompliceerder en moeten ze steeds sneller gereed zijn. Daarom is er behoefte aan technieken die het ontwikkelproces in goede banen leiden, onder meer met **systems engineering** en **capability maturity model integration**.

HANS VAN THIEL

Een embedded systeem is een product dat programmatuur, elektronica en meestal ook mechanica in zich verenigt. Daarmee valt het ontwerp van een embedded systeem onder de karakterisering van de Nederlandse afdeling van de International Council on Systems Engineering (Inco-se): *systems engineering* integreert alle disciplines en specialismen in een groepsaanpak en past een gestructureerd ontwikkelproces toe, van concept via productie tot en met operationeel gebruik.

Systems engineering bestrijkt meer dan alleen technische integratie. Inco-se definieert gebruik, prestatie, testen, productie, kosten & tijdschema, opleiding & ondersteuning en uitfaseren allemaal als SE-activiteiten. Hiermee wordt duidelijk vastgesteld dat ook projectmatige en organisatorische aspecten van een ontwikkelproces tot de discipline systems engineering (SE) behoren.

Ze zijn er natuurlijk wel, de kleine ontwikkelafdelingen die zich alleen met de techniek hoeven bezig te houden. Maar zodra productontwikkeling door honderden mensen en bovendien op verschillende locaties wordt gedaan is er een organisatie nodig met duidelijke (en gestructureerde) procedures.

System engineering raakt dan ook aan kwaliteitsbewaking en aan de diverse kwaliteitssystemen die in de loop der tijd zijn bedacht. En daarvan zijn er nogal wat. Behalve de ISO 9000 serie kennen we modellen van het European Foundation for Quality Management (EFQM) en de INK (Instituut voor Nederlandse Kwaliteit). Verder zijn er kwaliteitsstandaarden van de IEEE, EIA en ANSI, en combinaties van die standaarden.

In de software-industrie groeit met name de belangstelling voor de CMM-benadering (capability maturity model) van ontwikkelprocessen. Het SW-CMM is zelfs specifiek voor software bedacht door het Amerikaans Software Engineering Institute (SEI) en stamt al uit het begin van de jaren negentig. Later is het CMM-kader aangepast voor andere gebieden, waaronder systems engineering, en weer later is geprobeerd de diverse modellen te integreren in CMMI (capability maturity model integration, www.sei.cmu.edu/cmmi). CMMI bestrijkt vier deelgebieden: projectmanagement, procesmanagement, engineering en support. CMMI onderscheidt 24 procesgebieden, 194 *specific practices* en 22 *generic practices*.

Capability assessments

Eind vorig jaar organiseerden ADSE (Aircraft Development Systems Engineering) en MIS Ingenieurs Organisatie gezamenlijk een seminar onder de titel: *Productontwikkeling: procesverbetering en de rol van capability assessments*. ADSE is voortgekomen uit de luchtvaartindustrie (Fokker) en ADSE directeur Cheryl Atkinson was medeoprichter en voorzitter van de Nederlandse afdeling van Inco-se. MIS is gespecialiseerd in de organisatie van productontwikkeling en heeft als zodanig veel ervaring in de praktijk van CMM. Gastsprekers waren Bill Schoening van Boeing en Inco-se (zie ook kader), Ilse Langenberg van Hollandse Signaalapparaten en Wijbrand Medendorp van Ericsson Business Mobile Networks. De laatste twee sprekers behandelden met name hun ervaringen met Maturity-modellen binnen hun eigen organisaties.

CMM-modellen hebben als kenmerk dat de activiteiten worden onderverdeeld in verschillende categorieën en verschillende processen. Voor elk proces kan een stadium worden onderscheiden, met bijbehorend cijfer, en vaststellen in welk stadium het (be-

drijs)proces verkeert heet dan *capability assessment*. Uit al die verschillende cijfers kan een generiek cijfer worden samengesteld wat dan het CMM-niveau van de afdeling of het bedrijf voorstelt. Alle sprekers waren het er over eens dat een dergelijk getal niet als een rapportcijfer moet worden gebruikt; het ligt iets ingewikkelder. Derk Bol en Lion Wildenburg van ADSE slaagden erin dat duidelijk te maken door de zaal uit te nodigen tot een (beperkte) *capability assessment* van de eigen beroepsmatige activiteiten. De eerste activiteit (als voorbeeld) kwam uit de huishoudelijke sfeer. Hiervoor waren vier scores gedefinieerd.

Na het eten wordt alles afgewassen en opgeruimd.

Af en toe. Er wordt soms afgewassen, vaak ook niet.

Meestal, maar naar persoonlijk inzicht. Er wordt haast altijd afgewassen. Iedereen doet het op zijn of haar eigen manier.

Altijd en op per project afgesproken wijze. Er wordt afgewassen volgens een aan tafel afgesproken manier. Dit wordt gecontroleerd.

Altijd en afgeleid van een vaste methode. Er wordt afgewassen volgens aangepaste instructies van de huishoudschool.

Het voorbeeld toont goed aan dat hoger niet altijd beter is; niveau 2 zal in dit geval meestal voldoende zijn. Men kan zich voorstellen dat in de keuken van een groot ziekenhuis een hoog niveau wel van nut kan zijn.

Het volledige CMMI (zie kader) kent zes van deze *capability levels*, aangeduid met: *not performed, performed, managed, defined, quantitatively managed en optimizing*. (SW-CMM heeft overigens vijf niveaus in plaats van zes.) In de praktijk halen slechts enkele organisaties hogere niveaus dan level 4, zoals bijvoorbeeld de NASA.

'Het was een systeemfout'

'Eigenlijk gaat het om onzekerheden', zegt keynote speaker Bill Schoening, 'systems engineering houdt zich bezig met onzekerheden, niet in de stochastische betekenis, maar in de zin van onbekend, onbepaald (unidentified). Het ongeluk met de Ariane-raket was te wijten aan vijf fouten. Dat waren geen softwarefouten maar systeemfouten. De mogelijkheid van de opgetreden interacties was niet voorzien. Complexiteit is ook een vorm van onzekerheid.'

Bill Schoening is wiskundige van-huis-uit. Toen hij dertig jaar geleden bij Boeing, toen McDonnell Douglas, geïnteresseerd raakte in systems engineering bestond dat vakgebied nog niet.

Gevraagd naar wat systems engineering nu eigenlijk is, maakt hij een vergelijking met de wiskunde. Wiskunde bestaat uit algebra, topologie, statistiek en vele andere sub-disciplines en *systems engineering* bestaat ook uit zo'n groep deelgebieden. Er zijn twaalf verschillende rollen vastgesteld voor SE. Schoening verwijst naar enkele artikelen hierover, te vinden op www.software.org/pub/externalpapers. CMM en andere dergelijke modellen behoren als vakgebied ook tot SE.

Bij Boeing raakte men ervan overtuigd dat de oorzaak van belangrijke problemen tijdens ontwikkeling lag in verbroken of niet bestaande processen. Het meten van de *maturity* van deze processen zou dan wel eens een belangrijke bijdrage tot verbetering kunnen leveren.

Op dit ogenblik ziet Schoening het meest in CMMI dat met betrokkenheid van Incose is ontwikkeld en de specifieke CMM-modellen integreert. Overigens overlappen alle CMM's elkaar voor een groot deel. Zo verschilt software-ontwikkeling, volgens Schoening, niet zoveel van andersoortige productontwikkeling. Maar de systeem-ingenieur kijkt steeds naar de samenhang van de delen, naar het totaal.

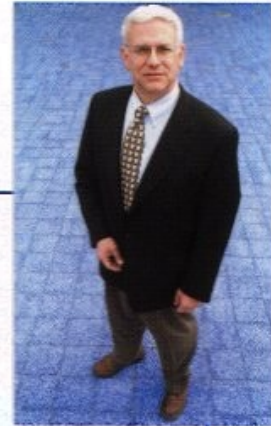
'Systems engineering is de lijm die het hele project bij elkaar houdt', aldus Schoening. Hij was in 1998 voorzitter van de International Council on Systems Engineering. Bij Incose heeft men een bestuursstructuur waarbij de voorzitter wordt bijgestaan door de voorganger en de opvolger, zodat Schoening dus drie jaar actief was op het hoogste niveau van die organisatie. Daarna H. Rhodes van Lucent is de huidige voorzitter.

Schoening rekende het tot een van zijn belangrijkste taken om het Amerikaanse instituut een meer internationaal karakter te geven en hij is er trots op dat dat - ook in de bijdragen aan congressen en werkgroepen - goed is gelukt. Overigens kent Incose ook een aantal landelijke afdelingen, waaronder een Nederlandse met op dit ogenblik zo'n 80 leden.

Op de internationale website is veel te vinden over *requirements*. Schoening beaamt dat binnen SE naar *requirements engineering* de meeste aandacht gaat. De kern van SE ligt echter niet in de afzonderlijke gebieden maar in de onderlinge wisselwerkingen.

Hij noemt een voorbeeld uit zijn eigen praktijk waarin een nieuwe versie van een COTS-processor (commercial off the shelf) werd geïntroduceerd die twee keer zo snel was als de vorige versie. De test-code van de applicatie was echter vijf keer zo traag. Uiteindelijk werd de oorzaak gevonden in een verder onschadelijke machine-instructie die een tienvoudige kopieeropdracht uitvoerde. De fout lag niet in de software en niet in de hardware, maar in de interactie van beiden. Het was een systeemfout.

'Nieuwe technologische uitvindingen berusten vaak op integratie', zegt Schoening. Soms bestaat innovatie uit één enkele technologische ontwikkeling maar meestal komt er een verbetering op een deelgebied die plotseling integratie met een aantal bestaande technieken mogelijk maakt. Schoening: 'Het wezen van systemen ligt in de synergie. Zonder synergie heb je geen systeem maar alleen een optelsom.'





De sprekers van ADSE vroegen de deelnemers uit de zaal zichzelf te beoordelen op aspecten van de volgende gebieden: *project monitoring en control, supplier agreement management, customer en product requirements, product integration, configuration management, process and product quality assurance, organizational process definition en causal analysis and resolution.*

Zo omvat product integration (van bijvoorbeeld een hard disk) het vaststellen en beheren van interfaces, het testen van de delen en van het geheel, en het ontwikkelen van een integratiestrategie bij de aanvang van de productontwikkeling.

De deelnemers aan het seminar konden - in realtime - door een knop in te

drukken, met een cijfer van een tot vier aangeven hoe bij hen dergelijke activiteiten werden afgehandeld. De samengestelde score van de zaal lag tussen de twee en drie, maar er waren behoorlijke verschillen tussen zowel de respondenten als de afzonderlijke processen binnen eenzelfde bedrijf.

De praktijk

De resultaten van een capability assessment, en zeker het samengestelde generieke cijfer, moeten met verstand worden geïnterpreteerd, zo benadrukten de deskundigen. Ten eerste is het niet altijd nodig het hoogste niveau na te streven, ten tweede zijn niet alle processen in alle situaties even belangrijk en tenslotte garandeert een gestructureerde en gecontroleerde werkwijze nog geen perfect resultaat.

Ilse Langenberg van HSA (tegenwoordig Thomson-CSF Signaal) stelde bijvoorbeeld dat bij het hanteren van CMMI alleen de van toepassing zijnde delen gebruikt moeten worden, bovendien op maat gesneden naar de specifieke behoeften van de organisatie. Het

model is een gids; het gaat er ook niet om een level meer te bereiken maar om de processen te verbeteren.

Wijbrand Medendorp van Ericsson gaf een uitgebreide analyse van de resultaten die wereldwijd bij dat concern zijn behaald. Systematisch ontwikkelen met gebruik van CMM heeft volgens hem aantoonbaar financiële voordelen opgeleverd. Medendorp's *bottom line* luidde: 'We kunnen nu op de afgesproken tijd en tegen een vooraf vastgestelde prijs een product ontwikkelen, en dat binnen normale werkweken.'

HVT

Achtergrond

www.adse.nl
www.mis.nl
www.incose.nl
www.incose.org
www.sei.cmu.edu
www.software.org/pub/externalpapers