

Evaluatie van softwarekwaliteitsconcepten en case-studie ervaringen

Citation for published version (APA):

Trienekens, J. J. M., & Eisinga, P. J. (1996). Evaluatie van softwarekwaliteitsconcepten en case-studie ervaringen. In C. Dam, van, & T. A. Beek, van (editors), *Bedrijfskundige uitdagingen en de manager van nu* (blz. 97-106). Kluwer Bedrijfswetenschappen.

Document status and date:

Gepubliceerd: 01/01/1996

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Evaluatie van softwarekwaliteitconcepten en case-studie ervaringen

Drs. P.J. Eisinga en dr.ir. J.J.M. Trienekens

De kwaliteit van softwareproducten is al jarenlang een min of meer ondefinieerbaar concept. Ontwikkelaars van software zijn nauwelijks in staat de kwaliteitskenmerken van hun softwareproducten in voor afnemers begrijpelijke taal te formuleren. Afnemers weten vaak niet welke kwaliteitseisen voor hun bedrijfssituaties realistisch zijn en welke verwachtingen zijn gerechtvaardigd.

In deze bijdrage wordt een model gepresenteerd voor de evaluatie van kwaliteitskenmerken van softwareproducten. De wensen en behoeften van afnemers van softwareproducten spelen daarin een centrale rol. Vervolgens wordt de toepassing van het evaluatiemodel in een case-studie beschreven. Voor de bedrijfskunde en voor het bedrijfsleven vormt het evaluatiemodel, en de op basis van het model te ontwikkelen evaluatiemethoden en -technieken, een uitdaging om 'fitness-for-use'-kwaliteitsbehoeften van praktijkmensen te concretiseren en te toetsen aan de technische kwaliteitskenmerken van softwareproducten.

Inleiding

Veel deskundigen in de softwarebranche, zowel onderzoekers als mensen uit de praktijk, hebben de jaren negentig bestempeld als het kwaliteitstijdperk. Afnemers van softwareproducten hebben kwaliteit bovenaan hun verlanglijstje gezet. Niet langer nemen zij genoegen met softwareproducten die 'enkel' aan functionele specificaties voldoen, op tijd worden geleverd en tegen redelijke kosten. Potentiële kopers van softwareproducten eisen dat de kwaliteit van software wordt geobjectiveerd en gekwantificeerd. De software-industrie is druk doende in deze behoeften te kunnen voorzien. Momenteel kunnen twee verschillende benaderingen van softwarekwaliteit worden onderscheiden. We noemen ze de procesbenadering en de produktbenadering.

Voorbeelden van de procesbenadering zijn ISO 9000-3 (1991), het 'Capability Maturity Model' (Humphrey, 1989) en het SPICE-project (Software Process Improvement Capability dEtermination) (Dorling, 1993). De procesbenadering kan worden beschouwd als een indirecte methode om te komen tot kwaliteitsverbetering van softwareproducten. De nadruk ligt op de beschrijving en de structurering van softwareprocessen. Afnemers van softwareproducten zijn nauwelijks betrokken bij deze benadering. Ondanks de interessante resultaten die werden geboekt, blijven veel vragen onbeantwoord:

- dienen ontwikkelaars alleen verantwoordelijk te zijn voor de kwaliteit van softwareproducten?
- wat zijn concreet de effecten van het verbeteren van ontwikkelprocessen op de kwaliteit van een softwareproduct?
- is 'technische' kwaliteit (bijvoorbeeld 'zero defects') hetzelfde als gebruikskwaliteit ('fitness-for-use')?

In deze bijdrage wordt gesteld dat het beantwoorden van dit soort vragen alleen mogelijk is via een produktbenadering van softwarekwaliteit. De produktbenadering heeft betrekking op de identificatie en specificatie van kwaliteitskenmerken van softwareproducten. Voorbeelden van kwaliteitskenmerken van softwareproducten zijn 'betrouwbaarheid', 'bruikbaarheid', 'efficiëntie' ('doelmatigheid'), 'onderhoudbaarheid' (zie ISO/IEC 9126, 1991). Voorbeelden van onderzoeksprojecten waarbinnen de produktbenadering centraal staat zijn op landelijk niveau het QUINT-project (SERC-QUINT, 1992) en op Europees niveau de ontwikkeling van ISO 9126-normen. Deze projecten richten zich met name op het uitwerken van kwaliteitskenmerken naar produkteigenschappen, en het definiëren van indicatoren en meetvoorschriften. Hoewel een bepaalde mate van objectiviteit en kwantificatie wordt bereikt voor wat betreft de realisatie van softwarekwaliteit, wordt vrijwel geen aandacht besteed aan de identificatie en de specificatie van kwaliteitskenmerken op basis van de behoeften van afnemers.

De bepaling van softwarekwaliteitskenmerken is reeds enkele jaren een onderzoeksthema binnen de Technische Universiteit Eindhoven (TUE). Concepten voor softwarekwaliteit werden ontwikkeld en in de praktijk gevalideerd, zie bijvoorbeeld Heemstra et al. (1993). Enkele van de belangrijkste conclusies uit dit onderzoek zijn:

- kwaliteitsbehoeften van afnemers kunnen worden gerelateerd aan kenmerken van bedrijfssystemen;
- softwarekwaliteitskenmerken kunnen worden geïdentificeerd door een analyse van de kenmerken van een bedrijfssituatie;
- bedrijfssituaties kunnen op gestructureerde wijze worden geanalyseerd om kwaliteitskenmerken van softwareproducten te identificeren en specificeren.

Deze inzichten dienen momenteel als basis voor een onderzoeks-project dat wordt uitgevoerd door KEMA Nederland BV. Doel is een methode te ontwikkelen voor een objectieve en onafhankelijke evaluatie van softwareproducten die is gebaseerd op internationale standaards (zoals ISO 9126). Door evaluatie van een softwareproduct moet de kwaliteit ervan zichtbaar en tastbaar worden gemaakt. Het uiteindelijke doel is het verbeteren van de kwaliteit van de softwareproducten die op de markt worden gebracht.

De structuur van deze bijdrage is als volgt. In paragraaf 1 worden de belangrijkste concepten ter bepaling van de kwaliteit van softwareproducten beschreven. In paragraaf 2 wordt ingegaan op de toepasbaarheid van deze concepten aan de hand van een case-studie. In paragraaf 3 worden de gemaakte vorderingen besproken en wordt aangegeven in welke richtingen het onderzoek wordt voortgezet.

1. Kwaliteit van softwareprodukten

In deze paragraaf wordt onderscheid gemaakt tussen kwaliteitsconcepten van softwareprodukten en concepten voor de evaluatie van de kwaliteit van softwareprodukten.

1.1 Kwaliteitsconcepten van softwareprodukten

Het softwarekwaliteitskenmerk

In 1977 werd door McCall et al. (1977) het voorstel gedaan om kwaliteit op te splitsen in een aantal kwaliteitsfactoren. Dit idee werd opgepakt door een groot aantal auteurs die hebben geprobeerd softwarekwaliteit onder te brengen in een verzameling kenmerken en subkenmerken die op hun beurt weer zijn gekoppeld aan metrieken. Op deze wijze zijn verschillende, hiërarchisch gelaagde modellen voor softwarekwaliteit beschreven. In deze modellen komen telkens dezelfde of soortgelijke kenmerken voor, hoewel de onderlinge rangorde in de hiërarchie steeds anders kan zijn.

Recent hebben de 'International Organization for Standardization' (ISO) en de 'International Electrotechnical Commission' (IEC) een norm voor softwarekwaliteitskenmerken gedefinieerd. Deze norm vormt een verdere stap op weg naar een consensus binnen de softwarebranche. Door ISO/IEC worden zes 'hoofd' kwaliteitskenmerken gedefinieerd en wordt voorgesteld ieder kwaliteitskenmerk onder te verdelen in een aantal subkenmerken. Deze kenmerken en de bijbehorende subkenmerken zijn respectievelijk:

- functionaliteit, bestaande uit vijf subkenmerken: geschiktheid (mate van afdekking van functies), nauwkeurigheid, koppelbaarheid, naleving (van standaards) en beveiliging;
- betrouwbaarheid, hetgeen kan worden onderverdeeld in de subkenmerken bedrijfszekerheid, bestendigheid en herstelbaarheid;
- bruikbaarheid, met de subkenmerken begrijpelijkheid, leerbaarheid en bedienbaarheid;
- doelmatigheid, onderverdeeld in tijdgedrag en middelgedrag;
- onderhoudbaarheid, bestaande uit vier subkenmerken: analyseerbaarheid, modificeerbaarheid (veranderingsvermogen), stabiliteit en testbaarheid;
- overdraagbaarheid (portabiliteit), bestaande uit vier subkenmerken: aanpasbaarheid, installeerbaarheid, conformiteit en vervangbaarheid.

In werkgroep zes (WG6) van ISO/JTC1/SC7 (1994) wordt doorlopend onderzoek verricht naar de definiëring en de bepaling van metrieken en normen voor elk van de afzonderlijke (sub)kenmerken.

Het kwaliteitsprofiel en het kwaliteitsniveau

Het vertrouwen dat afnemers in softwareprodukten stellen groeit naarmate evaluaties objectiever en grondiger kunnen worden uitgevoerd. Daarbij spreekt het voor zich dat niet alle (sub)kenmerken van elk softwareprodukt met dezelfde grondigheid hoeven te worden geëvalueerd. Zo is het bij een tekstverwerkingsprogramma bijvoorbeeld belangrijker de 'bruikbaarheid' ervan grondiger te onderzoeken dan de 'betrouwbaarheid' (hoewel dit niet inhoudt dat de 'betrouwbaarheid' van een tekstverwerkingsprogramma totaal onbelangrijk is). Voor de software die in een kerncentrale wordt gebruikt

geldt omgekeerd dat de 'betrouwbaarheid' ervan belangrijker is dan de 'bruikbaarheid' (maar ook dit betekent weer niet dat de 'bruikbaarheid' van geen belang is).

Dit soort intuïtieve stellingen zijn nader onderzocht in diverse internationale onderzoeksprojecten, waaronder het ESPRIT-project SCOPE (Software Certification Programme in Europe, zie Bache et al., 1994). Binnen SCOPE zijn richtlijnen opgesteld ter bepaling van de mate van grondigheid die nodig is bij de evaluatie van een bepaald softwareprodukt in een bepaalde bedrijfssituatie. Voorts zijn binnen SCOPE vier evaluatieniveaus opgesteld, met een oplopende mate van grondigheid (van niveau D (gering), niveau C (redelijk), niveau B (hoog) tot niveau A (zeer hoog)). De niveaus zijn opgesteld aan de hand van factoren zoals de economische risico's van uitval van een softwareprodukt. De SCOPE-richtlijnen zijn bedoeld als globale richtingaanwijzers die beoordelaars moeten helpen bij het bepalen van een softwarekwaliteitsprofiel. In een kwaliteitsprofiel worden de van toepassing zijnde (sub)kenmerken van kwaliteit en de bijbehorende evaluatieniveaus vastgesteld.

Het bovenstaande maakt duidelijk dat er concepten bestaan voor de kwaliteit van softwareproducten. Een vraag waar tot dusverre echter nog nauwelijks aandacht aan is besteed, is hoe softwarekwaliteitskenmerken kunnen worden geëvalueerd. Het ligt voor de hand dat afnemers zouden moeten proberen hun wensen en behoeften ten aanzien van softwarekwaliteit nader te omschrijven. Evenzeer ligt het voor de hand dat ontwikkelaars van software moeten proberen de kwaliteit van hun producten te kenmerken in termen die voor afnemers duidelijk zijn. Tot dusverre is er echter geen systematische methode beschikbaar waarmee deze activiteiten kunnen worden ondersteund of begeleid. De volgende paragraaf gaat nader in op concepten voor de evaluatie van softwarekwaliteit.

De evaluatie van de kwaliteit van een softwareprodukt

In onderzoeksprojecten van de Technische Universiteit Eindhoven (TUE) is recentelijk een grondslag ontwikkeld voor de identificatie en specificatie van softwarekwaliteitskenmerken (Trienekens, 1994). Deze houdt ondermeer in dat wensen, behoeften en eisen binnen bedrijfssituaties kunnen worden gerelateerd aan kwaliteitskenmerken van softwareproducten. Er wordt gebruik gemaakt van het zogenaamde Proces Besturing Informatie (PBI)-model (Bemelmans, 1986). Volgens het PBI-model kunnen de te evalueren kwaliteitskenmerken van een softwareprodukt worden bepaald door middel van een analyse van de kenmerken van bedrijfssituaties. De volgende paragraaf, 1.2, gaat nader in op deze benadering.

1.2 Van kenmerken van een bedrijfssituatie naar een softwarekwaliteitsprofiel

In de eerder genoemde TUE-benadering wordt onderscheid gemaakt tussen drie soorten kenmerken van bedrijfssituaties: bedrijfssysteemkenmerken, softwareproduktkenmerken en kenmerken van afnemers. Vervolgens wordt een model geïntroduceerd voor de evaluatie van softwarekwaliteit.

De bedrijfssysteemkenmerken

Het eerder genoemde PBI-model is gebruikt om de kenmerken van bedrijfssituaties te definiëren welke vervolgens in verband kunnen worden gebracht met softwarekwaliteitskenmerken. De bedrijfssituatiekenmerken die worden genoemd zijn respectievelijk:

- het *aantal* systeemvariabelen en hun *onderlinge afhankelijkheid*;
- de *beheersbaarheid* en *voorspelbaarheid* van systeemvariabelen;
- de *gevoeligheid* en *stabiliteit* van het bedrijfssysteem;
- het *reactiepatroon* van het bedrijfssysteem.

We geven een voorbeeld van relaties tussen dit soort bedrijfssituatiekenmerken en softwarekwaliteitskenmerken.

Naarmate het reactiepatroon van het bedrijfssysteem onbestendiger is, is de behoefte groter aan snel en doeltreffend actie kunnen ondernemen met het softwarepakket. Dit houdt ondermeer in dat hoge eisen worden gesteld aan het kwaliteits(sub)kenmerk 'tijdgedrag' van een softwareprodukt.

De softwareproduktkenmerken

Softwareproducten kunnen in drie dimensies worden gekenmerkt: de scope van een produkt, de fasen in de levenscyclus waarvoor de ontwikkelaar verantwoordelijk is en de uniekheid van een softwareprodukt (zie Trienekens).

- De *scope* van het produkt: omvat het produkt alleen de broncode, zijn de gebruikershandleidingen inbegrepen, of worden ook de inpassingsprocedures in de organisatie meegenomen?
- De *fasen* in de levenscyclus: is de ontwikkelaar verantwoordelijk voor zijn werk vóór aflevering of tijdens de gehele levenscyclus van een softwareprodukt?
- De *uniekheid* van het softwareprodukt: betreft het een standaard applicatieprogramma of gaat het om een uniek en op maat ontwikkeld softwareprodukt?

We geven een voorbeeld van de relaties tussen softwareprodukt-dimensies en kwaliteitskenmerken.

Indien de ontwikkelaar alleen verantwoordelijk is voor de fasen voor aflevering van een softwareprodukt, dan zal hij zich voornamelijk richten op de door de afnemer gewenste afdekking van functies (subkenmerk 'geschiktheid' van kwaliteitskenmerk 'functionaliteit'). Indien een ontwikkelaar verantwoordelijk is voor de totale levenscyclus van een softwareprodukt zal hij veel meer aandacht besteden aan kwaliteits(sub)kenmerken zoals 'bedrijfszekerheid', 'onderhoudbaarheid' en 'overdraagbaarheid'.

Ook menselijke aspecten spelen een rol bij de identificatie en specificatie van softwarekwaliteitskenmerken.

De afnemerkenmerken

Een afnemer kan direct dan wel indirect zijn betrokken bij de aankoop, het gebruik of beheer van een softwareprodukt. Afnemers kunnen worden onderverdeeld in de volgende twee categorieën:

- *Gebruikers*: zowel eindgebruikers als gebruikers op afstand (bijvoorbeeld management dat periodiek rapportages krijgt over de toepassing van een softwareprodukt);
- *Beheerders*: deze dragen respectievelijk de verantwoordelijkheid voor de werking van het softwareprodukt en verantwoordelijkheid voor de aanpassing van een softwareprodukt aan veranderende gebruikerswensen.

Deze afnemer-categorieën kunnen nader worden gespecificeerd aan de hand van respectievelijk de ervaring die afnemers hebben met vergelijkbare softwareproducten of met

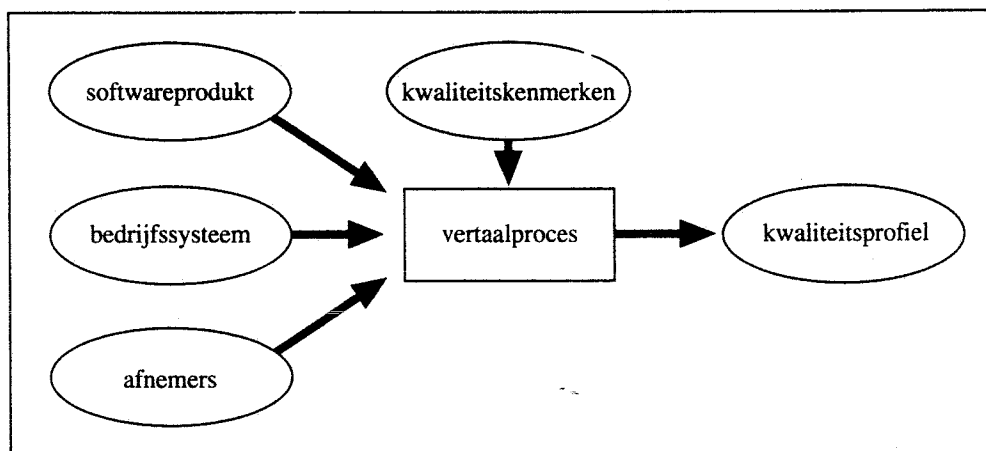
softwareproducten in het algemeen, het opleidingsniveau van afnemers en het aantal afnemers binnen een bedrijfssysteem. We geven een voorbeeld van afnemerkenmerken in relatie tot de bijbehorende softwarekwaliteitskenmerken.

Een softwareproduct dat is bedoeld voor een groot aantal onervaren, op invoer gerichte gebruikers met een laag opleidingsniveau, dient in hoge mate 'begrijpelijk' en 'leerbaar' te zijn (beide kenmerken zijn subkenmerken van het kwaliteitskenmerk 'bruikbaarheid').

Een model voor de evaluatie van softwarekwaliteit

Op basis van de in het voorgaande beschreven relaties tussen enerzijds kenmerken van softwareproducten, bedrijfssystemen en afnemers en anderzijds kwaliteitskenmerken van softwareproducten, wordt een 'kwaliteitsevaluatie'-model beschreven. Het doel is te komen tot een gestructureerd en herhaalbaar proces waarmee de behoeften van afnemers en hun bedrijfssituaties worden vertaald in kwaliteitskenmerken met bijbehorende evaluatieniveaus. Het model, dat hierna wordt toegepast in een case-study, is weer gegeven in figuur 1.

Figuur 1. Het kwaliteit-evaluatiemodel



2. De case-studie VISION

Het doel van de case-studie is de toepassing van het evaluatiemodel. Het resultaat is een kwaliteitsprofiel aan de hand waarvan de kwaliteit van het softwarepakket VISION kan worden geëvalueerd. Deze paragraaf eindigt met enkele concrete evaluatievoorbeelden.

2.1 Het softwareprodukt VISION

VISION is een softwareprodukt dat is ontwikkeld ter ondersteuning van de planning, het ontwerp en beheer van elektriciteitsnetten. Het softwareprodukt is een DOS-applicatieprogramma met een grafische gebruikersinterface. De broncode is geschreven in PASCAL. De doelgroep bestaat niet alleen uit distributiebedrijven van elektriciteit maar ook uit productiebedrijven en ingenieursbedrijven. Het softwareprodukt is afge-

stemd op de huidige behoeften van elektrotechnici (zowel specialisten als generalisten) bij het ontwerpen van elektriciteitsnetten. VISION biedt meer dan een verzameling rekenprogrammatuur. Het softwarepakket verschaft ook managementinformatie voor verschillende plannings- en beheerfuncties. Elke gebruiker van het pakket heeft toegang tot alle soorten informatie over netplanning. Het softwarepakket wordt up-to-date gehouden doordat de leverancier in elke volgende versie de nieuwste inzichten in de techniek van netplanning verwerkt.

2.2 Toepassing van het evaluatiemodel en beschrijving van het kwaliteitsprofiel

In het navolgende wordt het evaluatiemodel toegepast om het kwaliteitsprofiel van het pakket VISION te bepalen. Respectievelijk komen aan de orde de kenmerken van het softwareprodukt, de kenmerken van het bedrijfssysteem en de kenmerken van de afnemers van het softwarepakket.

De kenmerken van het softwareprodukt

Het softwareprodukt VISION kan met behulp van de drie produktdimensies scope, levenscyclusfasen en uniekheid (zoals eerder beschreven in subparagraaf 1.2) als volgt worden gekenmerkt.

- *Scope* van het produkt: naast de software op zich worden door de leverancier gebruikersprocedures en -handleidingen meegeleverd. De leverancier draagt ook zorg voor de inpassing van het softwareprodukt in de bedrijfsomgeving, bijvoorbeeld door middel van advies, opleidingen en service.
- *Fasen* in de levensduur: de fasen in de levenscyclus ná aflevering zijn van belang. De leverancier van het softwareprodukt heeft formeel de verantwoordelijkheid voor de verdere (versiegewijze) ontwikkeling.
- *Uniekheid*: het produkt is geen maatwerkpakket voor een specifieke afnemer. VISION kan worden gekarakteriseerd als een standaardpakket. Er bestaan in de markt alternatieve softwareprodukten.

Het bedrijfssysteem waarbinnen het softwareprodukt wordt toegepast

Het bedrijfssysteem in deze case-studie omvat de functies planning, ontwerp en beheer van elektriciteitsnetten. Voor de gebruikers is het niet het enige hulpmiddel. Ook andere bronnen van informatie zijn nodig. Toch kunnen fouten of storingen bij het gebruik van VISION aanzienlijke economische schade veroorzaken. Het bedrijfssysteem kan aan de hand van de kenmerken zoals beschreven in subparagraaf 1.2 als volgt worden beschreven.

- Het *aantal* en de *onderlinge afhankelijkheid* van systeemvariabelen: het aantal variabelen bij het plannen, ontwerpen en beheren van elektriciteitsnetten dat van belang is bij het toepassen van VISION wordt door deskundigen als gemiddeld ingeschat. De variabelen zijn sterk onderling afhankelijk.
- *Beheersbaarheid* en *voorspelbaarheid* van de systeemvariabelen: niet elke variabele bij het plannen, ontwerpen en beheren van elektriciteitsnetten laat zich eenvoudig beheersen of voorspellen.
- *Gevoeligheid* en *stabiliteit* van het bedrijfssysteem: de planning, het ontwerp en beheer van elektriciteitsnetten zijn geen gevoelige of instabiele bedrijfsfuncties.

De afnemers

De afnemers van het softwareprodukt zijn elektrotechnici binnen de energiedistributie- en -productiesector. De gebruikers en beheerders zijn veelal hoog geschoolde technici. Deze groep stelt steeds hogere eisen aan ondersteuning bij de planning en het ontwerp van elektriciteitsnetten.

Het kwaliteitsprofiel

Door het softwareprodukt, het bedrijfssysteem en het type afnemers te beschrijven konden de volgende ISO-kwaliteits-kenmerken als de meest relevante worden bepaald. (Dit wil uiteraard niet zeggen dat de overige kwaliteitskenmerken volstrekt onbelangrijk zijn.)

Functionaliteit

Functionaliteit (onder andere 'geschiktheid' (hier geïnterpreteerd als de dekking van de uit te voeren functies van het softwarepakket), en 'nauwkeurigheid'). Relevantie is hoog vanwege de:

- sterke onderlinge afhankelijkheid van de variabelen die een rol spelen bij de planning, het ontwerp en het beheer van elektriciteitsnetten;
- de economische schade die groot kan zijn bij onnauwkeurige informatieverstrekking door het softwarepakket;
- door gebruikers gewenste hoge mate van ondersteuning (lees: dekking) bij plannings- en ontwerpactiviteiten.

Bruikbaarheid

Bruikbaarheid (onder andere 'begrijpelijkheid', 'leerbaarheid'). Relevantie is hoog omdat:

- uiteenlopende groepen gebruikers (qua ervaring, opleidingsniveau) het softwarepakket gebruiken;
- het softwarepakket niet slechts een hulpmiddel is voor het uitvoeren van berekeningen aan elektriciteitsnetten maar ook planning, ontwerp en beheer van netten met diverse soorten managementinformatie moet ondersteunen.

Onderhoudbaarheid

Onderhoudbaarheid (onder andere 'modificeerbaarheid', 'testbaarheid'). Relevantie is hoog omdat:

- de leverancier zich heeft verplicht de verdere ontwikkeling van het softwareprodukt te garanderen;
- service tijdens gebruik van het softwareprodukt voor veel gebruikers uiterst belangrijk is;
- de levensduur van het softwareprodukt wordt geschat op 8-15 jaar.

De hoge relevantie van de genoemde kwaliteitskenmerken maakt het noodzakelijk deze kenmerken grondig te evalueren. In SCOPE worden voorstellen gedaan voor evalueren in verschillende maten van grondigheid aan de hand van indicatoren en meetvoor-schriften. In de navolgende paragraaf wordt kort ingegaan op het evalueren van de kwaliteitskenmerken van VISION.

2.3 Enkele voorbeelden van de evaluatie van het kwaliteitsprofiel van VISION

Bij de evaluatie wordt gebruik gemaakt van indicatoren en meetvoorschriften. We geven enkele voorbeelden daarvan bij elk van de in het voorgaande geselecteerde kwaliteitskenmerken van het softwarepakket.

Functionaliteit (de navolgende voorbeelden betreffen met name de 'geschiktheid' die hier wordt geïnterpreteerd als de mate waarin de gewenste functionaliteit wordt ondersteund door het softwarepakket).

Indicator: dekkingspercentage (gewenste functies/gerealiseerde functies).

Meting: de functionaliteit van VISION wordt vergeleken met een inventarisatie van de functies van vergelijkbare softwareproducten. Bijvoorbeeld wordt gekeken naar soorten netcomponenten en soorten invoer- en uitvoerfuncties.

Bruikbaarheid (de navolgende voorbeelden betreffen met name de 'begrijpelijkheid' van het softwarepakket).

Indicator: percentage toelichtende tekst.

Meting: er diende een inschatting te worden gegeven door een deskundige. Het softwarepakket is weinig behulpzaam, maar geeft voldoende hulp aan de gevorderde gebruiker door middel van ondermeer kleur in de statusbalk, oplichtende tekst in edit-schermjes, enzovoort. Een nadeel is de afwezigheid van een helpfunctie, bijvoorbeeld ontbreekt uitleg gegeven over de basisprincipes van netberekeningen.

Onderhoudbaarheid (de navolgende voorbeelden betreffen met name de 'modificeerbaarheid' van het softwarepakket).

Indicator: wijzigingsinspanning per eenheid van omvang van het softwarepakket.

Meting: voornamelijk aan de hand van ervaringen in de praktijk worden uren en doorlooptijd bepaald.

3. Conclusies en verder onderzoek

In de case-studie vormde het kwaliteitsevaluatiemodel een nuttig hulpmiddel voor de beschrijving van het kwaliteitsprofiel van het softwareprodukt VISION. Aan de hand van de drie invalshoeken softwareprodukt, bedrijfssysteem en gebruikers konden de meest relevante kwaliteitskenmerken worden bepaald. Het kwaliteitsprofiel vormt vervolgens de basis voor de evaluatie van de kwaliteit van het softwareprodukt.

Verder onderzoek zal zich richten op respectievelijk de verdere uitwerking van het kwaliteitsevaluatiemodel, de evaluatieniveaus en de verdere concretisering van de evaluatie van een softwareprodukt. We lichten dit kort toe.

Uitwerking van het kwaliteitsevaluatiemodel.

Voor elk van de drie invalshoeken softwareprodukt, bedrijfssysteem, gebruikers, wordt een vragenlijst ontwikkeld die aan afnemers kan worden voorgelegd en waarmee aan de hand van hun antwoorden een prioriteitenstelling van kwaliteitskenmerken kan worden afgeleid.

Normering van evaluatieniveaus.

Het hanteren van evaluatieniveaus vraagt om normwaarden. Door het ontbreken hiervan moest het bepalen van evaluatieniveaus in de VISION-casus noodgedwongen beperkt blijven tot een subjectieve inschatting van de mate van grondigheid van de evaluatie van elk van de geselecteerde kwaliteitskenmerken. Verder onderzoek richt zich op de ontwikkeling van objectieve normen voor de evaluatieniveaus.

Concretisering van de evaluatie van softwarekwaliteit.

Vooralsnog bestaat er een beperkt inzicht in de evaluatiemogelijkheden van softwareproducten, met name de verbanden tussen soorten softwareproducten, kwaliteitskenmerken en indicatoren. In het onderzoekproject zullen relaties tussen indicatoren en kwaliteitskenmerken worden beschreven en geconcretiseerd. Er wordt gestreefd naar de ontwikkeling van een typologie van softwareproducten. Per type softwareproduct zullen vervolgens relevante kwaliteitskenmerken en bijpassende indicatoren en meetvoorschriften worden bepaald. Het uiteindelijke doel is een objectief en efficiënt evaluatieproces van de kwaliteit van softwareproducten.

Het ligt in de bedoeling dat de in het bovenstaande beschreven onderzoeksactiviteiten op relatief korte termijn zullen leiden tot een goed onderbouwde maar praktische methode voor een afnemer-georiënteerde evaluatie van softwareproducten. De uitdagingen voor bedrijfskunde en bedrijfsleven betreft met name de objectivering en kwantificering van de kwaliteit van softwareproducten vanuit een bedrijfskundig perspectief.

Literatuur

- Bache R. en G. Pazzana, *Software metrics for product assessment*, McGraw-Hill, London, 1994.
- Basili V.R. en J.D. Musa, 'The future engineering of software: a management perspective', in: *IEEE Computer*, Vol. 24, Nr. 9, 1991.
- Bemelmans, T.M.A., 'Bedrijfskundig Ontwerpen van Bestuurlijke Informatiesystemen', in: P.A. Cornelis, J.M. van Oorschot (red.), *Automatisering met een menselijk gezicht*, Kluwer, 1986.
- Boehm B.W. et al, 'Characteristics of Software Quality', in: *TRW Series of Software Technology*, Vol. 1, North-Holland, 1978.
- Davis G.B. en M.H. Olsen, *Management Information Systems*, McGraw-Hill, London, 1984.
- Delen, G.P.A.J. en D.B.B. Rijsenbrij, 'The Specification, Engineering and Measurement of Information Systems Quality', in: *Journal System Software*, Vol. 17, 1992.
- Dorling A., 'SPICE: Software Process Improvement and Capability determination', in: *Software Quality Journal*, Vol. 2, 1993.
- Heemstra F.J., R.J. Kusters en J.J.M. Trienekens, 'Defining Systems Quality: involving end-users', in: *Proceedings of the Conference of Function Point Users*, Bristol, Engeland, 1993.
- Humphrey W.S., *Managing the Software Process*, Addison-Wesley, 1989.
- Information Technology Security Evaluation Criteria (ITSEC), *Provisional harmonised criteria*, version 1.2. Commissie van de Europese Gemeenschappen, 1991.
- ISO 9000-3, *Quality management and quality assurance standards, Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software*, International Organization of Standardization, 1991.
- ISO/IEC 9126, *Information Technology - Software product evaluation - Quality characteristics and guidelines for their use*, International Organization of Standardization, 1991.
- ISO/IEC JTC1/SC7 N1201, *Working Draft; Information Technology: Indicators and Measures*, ISO/IEC, 1994.
- Kaposi A. en M. Myers, *Systems, models and measures*, Springer-Verlag, London, 1994.
- McCall, J.A., P.K. Richards en G.F. Walters, *Factors in Software Quality*, RADC-TR-77-363 Rome Air Development Center, Griffis Air Force, Rome, NY, 1977.
- SERC-Quint, *De specificatie van softwarekwaliteit, een praktijkhandleiding*, Kluwer Bedrijfswetenschappen, 1992.
- Trienekens J.J.M., *Tijd voor kwaliteit, werken aan betere informatiesystemen*, Thesis Publishers, Amsterdam, 1994.