

DOCUMENTATIE IN DE AUTOMATISERINGSCONTROLE

door J. H. J. Achten

1. Inleiding

Er is weinig fantasie voor nodig om het belang van een adequate documentatie van geautomatiseerde systemen in te zien: Beheersing van een proces vereist minimaal kennis van dat proces. Het is dan ook niet verwonderlijk dat in de literatuur in alle toonaarden de noodzaak van een goede documentatie wordt onderstreept.¹⁾

Maar juist daarom is het zo verbazingwekkend, dat bij praktische toepassingen van systemen die documentatie vaak zo bedroevend slecht is. Als voorbeeld wil ik noemen een financiële huishouding die in totaal ca 1.600 grotere en kleinere programma's in gebruik had. Van ongeveer de helft was de documentatie volstrekt onvoldoende. Inlopen van de achterstand werd op ca f 1 miljoen kosten geschat, nog afgezien van de schade doordat de programmeurs gedurende deze bezigheden niet aan nieuwe programma's zouden kunnen werken. Voor dit bedrijf met een jaarresultaat van enkele miljoenen, was een dergelijke aderlating onaanvaardbaar.

Niet overal waren de ervaringen zo slecht. Maar veelal was het niveau onvoldoende, was om allerlei redenen de documentatie een lage prioriteit gegeven.

In dit artikel wordt getracht het waarom van deze lage prioriteit te vinden: als documentatie voor het bedrijf zo'n noodzaak is, waarom is het bedrijf niet zelf de gene die daarvoor zorgt?

2. Het belang van documentatie I

Het is nuttig eerst het belang van documentatie tegen de bestaande controle-opvattingen aan te stippen.

In de controle van geautomatiseerde systemen is adequate documentatie een onmisbare schakel. Of het een nieuw programma dan wel een wijziging betreft: de beschrijving moet vastliggen, de gebruiker (wie dat ook moge zijn) autoriseert die beschrijving, hij draagt testgevallen aan om het programma te testen en bekijkt de resultaten om te zien of alles in het programma zit wat er in moet zitten, hij kijkt of er niet méér in het programma zit dan er in moet zitten (combi!) en na zijn zegen gaat het programma in de operationele bibliotheek, buiten bereik van de programmeurs.

¹⁾ Als literatuur is te noemen:

- Automatisering en controle (Nivra-geschriften 1/13), deel I, hoofdstuk IVA, deel III, hoofdstuk IV-3.
- Computer audit guidelines (CICA), sectie C.
- Leerboeken informatica; Methoden voor systeemonderzoek (Hartman en Roos), paragraaf 4.4.
- Bestuurlijke informatietechnologie (Starreveld), hoofdstuk XXII, paragraaf 10.
- Automatisering van de informatieverzorging (Van Belkum en Van 't Klooster), paragrafen 2.4, 3.2, 4.2-2 e.v.
- Auditing: an integrated approach (Arens en Loebbecke), pagina 384.
- Auditing principles (Stettler), pagina 525.
- Louter impliciet in Handboek accountancy, hoofdstuk III-30: De betekenis van de automatisering voor de accountantscontrole (Drent), paragraaf 2, met name pagina 3607.

Er is nogal wat te documenteren. Systeemontwerp en programmabeschrijving zijn de belangrijkste. Als controle-schakel is de programma-beschrijving onmisbaar, omdat deze alle belangrijke stappen in voor de gebruiker toegankelijke vorm moet weergeven.

3. De filosofie werkt vaak niet

De praktische uitwerking van deze filosofie blijkt vaak een ontgoocheling: Slechte of soms zelfs geen documentatie, geen of weinig participatie van de gebruiker, laat staan testen en autoriseren door deze. Navraag naar de oorzaken levert redenen op als de volgende:

- Documentatie is duur. Programmeurs schatten al snel dat een goede documentatie 20 à 25% van hun tijd opslokt. Bedrijven zijn niet genegen deze kosten te dragen of kunnen dat in deze sombere tijden niet.
- Programmeurstijd is schaars, een goede documentatie maakt deze nog schaarser en het gevaar van vertraging in automatiseringsprojecten doemt op.
- Gebruikers zien vaak niet de noodzaak van participatie in of zij worden afgeschrikt doordat deze wereld zo ver van hun deskundigheid af staat.

Het is niet gezegd dat daarmee alle redenen genoemd zijn. De opsomming heeft een sterk gecatalogiseerd karakter: Het kan allemaal wel zo zijn, maar daarmee is nog niet de invloed van die oorzaken vastgesteld. Of fundamenteeler gesteld: Daarmee is niet de oorzaak geanalyseerd.

In de volgende paragrafen wordt geprobeerd door middel van een andere wijze van benadering tot een meer werkbare oplossing te komen.

4. Het belang van documentatie II

Tot nu toe werd het belang van de documentatie vooral gezien als een schakel via welke de gebruiker een greep kan houden op een juiste verwerking van zijn gegevens.

Er is evenwel ook een ander belang en wel dat van een adequate werking van de diverse automatiseringsafdelingen:

- Systeemanalyse en programmering: Zonder een goede kennis van een systeem wordt het aanpassen aan gewijzigde omstandigheden en wensen al gauw onmogelijk zonder direct tot een geheel nieuw (sub-)systeem te moeten komen.
- Operating: De operateurs behoeven kennis omtrent de werking van een systeem om hun taak uit te kunnen voeren.
- Systeembeheer: Zonder een goede documentatie is het voor deze afdeling ondoenlijk greep te houden op de juiste werking van de juiste programma's.

De documentatie die doorgaans voorhanden is, betreft (van achteren naar voren): vrijwel steeds source-listing gemaakt tijdens compilatie van het bronprogramma naar het operationele programma, meestal een systeemontwerp o.d. van een globaal karakter, minder vaak vastlegging van onderzoek en besluitvorming. Bij een iets meer geavanceerd systeem is de programma bibliotheek vaak geautomatiseerd (Librarian e.d.); en daarmee is een hulpmiddel verkregen om aansluiting te vinden tussen operationele programma's en de laatste versie van de source-listing: als versienummer wordt het compilatietijdstip gebruikt.

5. De controlefilosofie nader bezien

Om tot een oplossing van het documentatieprobleem te komen volgt hier eerst een kritische beschouwing van de controle-filosofie zoals hiervoor vermeld onder „Het belang van documentatie I”.

Die benadering lijkt op het eerste oog sterk analytisch: We hebben nog nergens de externe controleur actief laten optreden; beoordeling van de interne organisatie staat voorop; de werking van de interne organisatie kan beoordeeld worden door inbreng van gebruikers na te gaan, door testresultaten te beoordelen, aangevuld met technische hulpmiddelen als audit packages. De controle-filosofie is dan ook vooral een externe controle filosofie. Maar ook vanuit die externe controle bezien is relativering nodig:

- Nagaan inbreng gebruikers: Deze inbreng is altijd essentieel: de gebruiker móét weten dat wat met zijn cijfers gebeurt, volgens de te stellen eisen gebeurt. Maar het is de vraag of deze inbreng zo eenvoudig kan worden waargenomen door een externe controleur die bij het proces van deze inbreng nauwelijks aanwezig kan zijn. Aan de hand van notulen en formele autorisaties is achteraf weliswaar enige inbreng te constateren, maar de waardering van deze inbreng is achteraf zeer moeilijk: Bijvoorbeeld vaststellen of een autorisatie louter een formele kwestie is, valt na het proces waarin dat gebeurde, maar zeer matig te zien.
- Beoordeling test-resultaten: Wie nog mocht menen dat een programma volledig te testen is, leze Tom Gilb's „Betrouwbaarheid en het ontwerpen van informatiesystemen” om uit die droom te raken. Bovendien, hoe is achteraf vast te stellen dat de testresultaten zijn geproduceerd door het nu operationele programma. Het alternatief om dan maar de source-listing te bekijken, geeft ook geen bevredigende oplossing omdat de externe controleur deze slechts achteraf kan bekijken: uit een operationeel programma valt nu eenmaal niet een source-listing terug te vertalen.
- Audit packages: Deze zijn stellig van belang, maar zij moeten niet worden overgewaardeerd. Het is zinvol onderscheid te maken tussen audit packages gericht op de programma's en die gericht op de gegevensverzamelingen:
 - = T.a.v. programma's: Aannemende dat een integrated test facility voor operationele programma's beschikbaar is, dat we kunnen testen wat we willen; dan nog blijft de vraag of, nadat de externe controleur zijn hielen heeft gelicht, dit programma en niet een ander gebruikt wordt.
 - = T.a.v. gegevensverzamelingen: Het zal normaliter zeker mogelijk zijn om gegevensverzamelingen op magnetisch medium, te verifiëren met een post uit balans of grootboek, om ter controle van het bestaan der gegevens een aantal te selecteren en deze te verifiëren. Maar is dat nog automatiseringscontrole? De gedachtengang van bepaalde auteurs²⁾ volgend, bepaald wel: Alleen naar het proces kijken is ondoelmatig, we moeten tevens de resultaten van dat proces nagaan. Deze noodzaak staat voor mij buiten twijfel: als een proces goed is dan moeten ook de resultaten goed zijn. Het omgekeerde geldt echter alleen als aangetoond kan worden dat een goed resultaat zich alleen kan voordoen bij een goed proces. En deze voorwaarde zal niet altijd vervuld zijn³⁾.

²⁾ Zie met name Blokdijk: „Een kernvraagstuk van de leer der accountantscontrole” (MAB 1975).

³⁾ Voor een fundamentele benadering van de band tussen hypothese en toetsresultaat, zie Klant's „Spelregels voor economen”.

- Een niet opgesomd hulpmiddel dat nog wel in de literatuur genoemd wordt (zie bijv. het controleprogramma van A. ten Pleet in „Districol en Minimax”) is: onderzoek logging. Nog afgezien van wat onderzoek achteraf van deze gegevens waard is, vereist dit een dusdanig technische kennis dat uitvoering van dit onderzoek door een doorsnee-automatiseringscontroleur illusoir is. Al deze auteurs zouden zelf eens zo'n poging moeten doen!

Uit deze beschouwing komt steeds naar voren dat controle op het automatiseringsproces verder reikt dan externe controle en zij is dan ook niet door externe controle vervangbaar. Of we willen of niet, voor deze controle zullen we ons meer op de interne organisatie en controle moeten verlaten.

In de volgende paragraaf is een andere benadering meer in haar algemeenheid uitgewerkt, daarna volgt een meer specifieke behandeling van de documentatie.

6. Interne organisatie van de automatisering

De interne organisatie van de sector organisatie in een bedrijf, gericht op de betrouwbaarheid van systemen en gegevensverzamelingen, gaat steeds uit van functionele onafhankelijkheid tussen enerzijds systeemanalyse en programmering, anderzijds uitvoering. Deze scheiding behoeft aanvulling met maatregelen als: Geen toegang van programmeurs etc. tot operationele programma's, geen detailinformatie aan operateurs inzake operationele programma's.

De scheiding tussen programmering en uitvoering moet verder worden aangevuld met een derde onafhankelijke functie, nodig om de scheiding tussen de eerste twee effectief te kunnen maken. Oorspronkelijk is hierbij steeds aan een bewaringsfunctie gedacht. Het bewaren betrof programma's en gegevensverzamelingen, deze zouden slechts worden afgegeven aan geautoriseerde personen, liefst tegen kwijting.

Naarmate invoer en verwerking van gegevens steeds meer on-line ging geschieden, kwam deze bewaringsfunctie in het gedrang: afgifte van programma's en bestanden hoeft niet meer, die staan reeds aangesloten op het on-line systeem.

Het is in deze situatie dat de behoefte groeit aan een functie systeembeheer en softwarecontrole: Die waakt over juist gebruik van juiste programma's en gegevensverzamelingen. Bij deze functie gaan de belangen van interne en externe controle sterk parallel lopen: Zonder deze interne organisatie zijn bedrijfsleiding en gebruikers niet in staat de betrouwbaarheid van de automatisering in het bedrijf te beoordelen, en kan de externe controleur dit ook niet.

De functie systeembeheer en softwarecontrole kan een belangrijke rol spelen met betrekking tot de documentatie, zowel de zorg dat deze adequaat is als dat deze gebruikt wordt. Zij is in staat documentatie te hanteren die bij programmeurs zelf in gebruik is, afzonderlijke vertaling is niet nodig.

Een opsomming van de taken die deze functie zou moeten inhouden, is tamelijk extensief - de volgende:

- Zorg dat de gebruikers vanaf het eerste stadium voldoende invloed op het nieuwe of herziene systeem hebben.
- Vaststellen dat een systeemontwerp aan de gebruikers verstrekt wordt, dat deze alle consequenties overzien en daarmee instemmen.

- In de testfase gebruikers proberen te betrekken bij het testen en deze zelf testgevallen laten aandragen. Indien dit niet (voldoende) lukt, als compensatie zelf intensiever met de test bemoeien.
- In de testfase zelf vaststellen dat alle kritische punten in het programma uitgeprobeerd worden, ook hier testresultaten beoordelen.
- Na compilatie van bron-programma naar operationeel programma, de source-listing met versienummer als compilatietijdstip vergelijken met het systeemontwerp om te zien of beide overeenstemmen.
- Op onverwachte ogenblikken het computercentrum binnenstappen en zien of de juiste programma's en bestanden (volgens een zekere werkvoorbereiding) worden afgewerkt: de logging krijgt nu opnieuw betekenis! Persoonlijke indrukken hoe makkelijk het soms schijnt te zijn om in een computercentrum privé-jobs (bijv. adressenbestand van een vereniging) te verwerken, doen de noodzaak van deze taak inzien.
- In geval van break down van de computer en andere calamiteiten: Proberen bij het herstel aanwezig te zijn, zich een beeld te verschaffen van het gebeurde, hier een verslag van te maken. Het belang van deze activiteit is niet in de laatste plaats omdat bij calamiteit zo makkelijk het stramien van de functiescheiding doorbroken wordt doordat alle functies een gelijkgericht belang hebben de machine weer „aan de praat” te krijgen.

Deze benadering is ook voor de externe controle bruikbaar, want deze heeft met deze activiteiten meer mogelijkheden inzicht te krijgen in de werking van de interne organisatie. Daar het activiteiten betreft die verder reiken dan de techniek van de externe controle, is deze aanpak effectiever dan te proberen door eigen waarneming vast te stellen of programma's bevatten wat ze moeten inhouden en niet meer.

De activiteiten van een functie systeembeheer en softwarecontrole zijn niet uitgediept. Daarom behoeft het documentatie-aspect - waar dit stuk toch over gaat - nog nadere uitwerking.

7. Tot slot opnieuw de documentatie

Ook in de benadering vermeld in de vorige paragraaf, is de documentatie een onmisbare schakel: zonder deze kunnen controles niet worden uitgevoerd of tenminste niet worden waargenomen. In dat opzicht is er geen verschil met de eerder vermelde controle-filosofie.

Dat onderscheid betreft andere aspecten:

- De automatiseringsafdelingen van het bedrijf kunnen werken met documentatie die aansluit bij de programma's zelf. Direct is te zien (bijv. ingeval van calamiteit) wat de beperktheden van een programma zijn en kan daar antwoord op gegeven worden.
- Aansluiting bij de behoeften van het bedrijf. Dit heeft immers een groot belang in aanwezigheid van documentatie die zo actueel mogelijk is. Documentatie die aanwezig is ten behoeve van de automatiseringsafdelingen zelf, zal daar eerder aan voldoen dan speciaal als verantwoording gemaakte documentatie.
- Betere inbreng van de gebruikers. Wie de gebruikers ook zijn mogen, het is een heterogene groep. Mensen met soms in het geheel geen gevoel voor au-

tomatisering, anderen die zich redelijk in de materie hebben verdiept maar geen partij zijn tegenover automatiseringsdeskundigen, weer anderen die je op dit gebied niets hoeft wijs te maken, zij allen vormen de groep die „gebruikers” heet en voor wie documentatie bestemd is. Een functie in het bedrijf die een brug weet te slaan tussen deze heterogene groep en wat er in feite met hun cijfers gebeurt, is wel nuttig.

- Kosten: Ofschoon de geschetste functie systeembeheer en softwarecontrole in een actieve rol zeker méér zal kosten, staan daar ook kostenvoordelen tegenover. Afzonderlijke documentatie als verantwoording kost dure programmeurstijd, zowel gerekend in echte kosten als in opportunity costs (de schade wegens het niet kunnen programmeren gedurende die tijd).
- Realisme in de controlemogelijkheden. Met name de externe controleur is beter in staat waar te nemen hoe de functie systeembeheer en softwarecontrole werkt en deze functie is beter in staat in de behoeften van de extern controleur te voorzien dan dat die externe controleur door eigen waarnemingen situaties probeert te beoordelen die hij eigenlijk alleen door continue bewaking onder controle zou kunnen houden.

Inmiddels zal zijn opgevallen dat de benodigde documentatie niet gering is. Behalve systeemontwerp en source-listing en de controlevastlegging dat beide overeenstemmen, zijn verslagen nodig betreffende inbreng van de gebruikers, instemming van hen met het systeemontwerp, is nodig documentatie van het testen, zijn verslagen nodig van overvalscontroles, break downs etc.

Maar het is documentatie die niet louter dient om gebruikers en controleurs tevreden te stellen maar die met alle processen verweven is. Het belang van het bedrijf is evident, het afdwingen (bijv. door een extern accountant) van adequate documentatie zal daarom op minder weerstanden stuiten en de documentatie zal minder een stiefkind zijn.

8. Slotbeschouwing

In dit artikel is niet getracht het belang van documentatie te relativieren, integendeel. Maar als documentatie zo vaak gebrekkig blijkt te zijn, dan is er reden te zoeken naar andere oorzaken dan die meestal verondersteld worden.

In zekere zin ben ik op het spoor gezet door een uitspraak van De Mare die waarschuwde voor informatie die niet echt door het bedrijf wordt gebruikt. „Dat soort informatie denatureert”, zei hij. Als dat gevaar in het algemeen aanwezig is, waarom dan niet bij slecht-gebruikte documentatie.

Het alternatief ligt dan voor de hand: bouw dit aspect uit de keten van interne organisatie en -controle in in de totstandkoming en verdere ontwikkeling van het automatiseringsproces in het bedrijf. Hiermee zijn zowel de belangen van het bedrijf zelf als die van de externe controle gediend.