

AC Adm. zaken
Dossier exemplaar

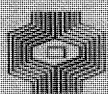
compact

COMPUTER EN ACCOUNTANT

UIT DE INHOUD

- o ACCOUNTANT, CONTINUITEIT, AUTOMATISERING EN RISICO-ANALYSE 2

- o PROGRAMMACONVERSIE - SYSTEEMONTWIKKELING MET HANDICAP 11



Klynveld Kraayenhof & co
ACCOUNTANTS



NUMMER 24

8E JAARGANG

ZOMER 1981

INHOUDSOPGAVE :

- o ACCOUNTANT, CONTINUITEIT, AUTOMATISERING
EN RISICO-ANALYSE
DOOR DRS. H.C. KOCKS 2
- o PROGRAMMACONVERSIE - SYSTEEMONTWIKKELING
MET HANDICAP
DOOR J.C.P.M. VERMEEREN 11
- o REISVERSLAG USA EN CANADA
DOOR A.H.C. KOEDIJK 22
- o COMPUTERTOEPASSING : VOORRAADWAARDERING
DOOR DRS. J.E. HUIZENGA 30
- o DE BETEKENIS VAN DE PROGRAMMEURSSECTIE VOOR
AUTOMATISERING EN CONTROLE
DOOR H. WEERD 35
- o BOEKEN
RUBRIEKSREDACTEUR J. PHILIPPG 39
- o LITERATUUR
RUBRIEKSREDACTEUREN J.C.P.M. VERMEEREN
EN DRS. B.M. DE VRIES 41
- o A.B.C.-NIEUWS
RUBRIEKSREDACTEUREN J.F.C. VAN EPEN EN
DRS. H.C. KOCKS 47
- o EXTERNE CURSUSSEN : ONDERWIJS
DOOR H. LAFEBRE 67

NUMMER 24

8E JAARGANG

ZOMER 1981

VAN DE REDACTIE

=====

Het zomernummer 1981 van Compact geeft het tweede en afsluitende artikel van drs H.C. Kocks over accountant, continuïteit, automatisering en risicoanalyse. Het is verheugend dat wij thans over een aanzet tot meningsvorming over dit artikel beschikken.

De redactie is zeer benieuwd naar reacties uit de algemene controlesector. Naar verwacht kan vooral commentaar worden tegemoet gezien met betrekking tot het gedeelte dat handelt over Accountant en continuïteit.

Ook het tweede hoofdartikel vraagt de bijzondere aandacht van de lezer gezien de voortdurende beweging in de automatisering waardoor programmaconversie een veel voorkomend verschijnsel is.

Evenals voorheen ziet de redactie reacties van de lezers met belangstelling tegemoet.

Compact is een uitgave van de Automatisering en Controle Groep van Klynveld Kraayenhof & Co..

Het doel van deze uitgave is informatie te verstrekken over ontwikkelingen op het gebied van automatisering en controle in binnen- en buitenland.

Deze informatie is in de eerste plaats bestemd voor diegenen, die in de algemene controlepraktijk werkzaam zijn.

Redactie:

Drs. J.E. Huizenga,
A.W. Neisingh en
Prof. D. Steeman.

Kopij kunt U inleveren bij de secretaris van de redactie:

H.J.M. van der Wielen.

Adres:

Pr. Irenestraat 59,
1077 WV Amsterdam.

Postadres:

Postbus 7137
1007 JC Amsterdam.

door drs. H.C. Kocks

Inleiding

In het vorige compactnummer is uiteengezet waarom de accountant aandacht zou dienen te besteden aan de niet-financiële factoren die de continuïteit van de onderneming zouden kunnen bedreigen. Daarna is de problematiek toegespitst op de automatisering. Aan het slot is aangegeven welke risico's de accountant loopt indien hij in daarvoor in aanmerking komende gevallen geen of te weinig aandacht aan die problematiek schenkt.

In deze aflevering zal de bepaling en de uitwerking van de continuïteitsafhankelijkheidsgraad (cag) nader uiteen worden gezet. Onder de bepaling van de cag wordt verstaan het aangeven van de mate waarin de continuïteit van de onderneming afhankelijk is van de automatisering. De uitwerking ervan tracht de vraag te beantwoorden of bij geconstateerde afhankelijkheid de getroffen beveiligingsmaatregelen voldoende zijn om de continuïteit van de geautomatiseerde gegevensverwerking en informatieverstrekking in redelijke mate te kunnen waarborgen.

Het gaat er in het volgende niet om een methode te beschrijven. Het is de bedoeling een benaderingswijze uit de doeken te doen, die de problematiek van de beveiliging met betrekking tot de automatisering op een andere wijze benadert dan tot nu toe in de literatuur het geval is geweest; namelijk vanuit de bedrijfsfunctie en niet vanuit de automatisering.

5.0 Uitwerking

5.1 Uitgangspunten

Zoals reeds in de inleiding is gezegd kan van een functiegewijze benadering worden gesproken. Wat houdt dat nu in. Er wordt vanuit gegaan, dat een organisatie als een samenstel van functies kan worden gezien, die elk een bepaalde betekenis hebben voor de continuïteit van de bedrijfsactiviteiten. De ene functie kan echter als vitaler worden beschouwd dan de ander. Dit houdt in dat het uitvallen van die ene functie eerder - en meestal ernstiger - gevolgen zal hebben voor de continuïteit van de bedrijfsactiviteiten. Daarom zal, indien functies gebruik maken van de automatisering, de ene functie hogere beveiligingseisen stellen met betrekking tot automatisering dan de andere. Anders gezegd: de vitaliteit van de functie bepaalt de vereiste mate van beveiliging met betrekking tot de automatisering.

Verder wordt ervan uitgegaan dat:

- een functie kan bestaan uit twee of meer subfuncties waarbij elke subfunctie slechts deel kan uitmaken van één functie. (functie: subfunctie = 1 : N);

- binnen een organisatie twee of meer hiërarchische niveaus zijn te onderkennen, waarbij twee niveaus als relatie hebben functie: subfunctie. (niveau N : niveau N + 1 = functie : subfunctie).

Schematisch is het voorgaande als volgt weer te geven:

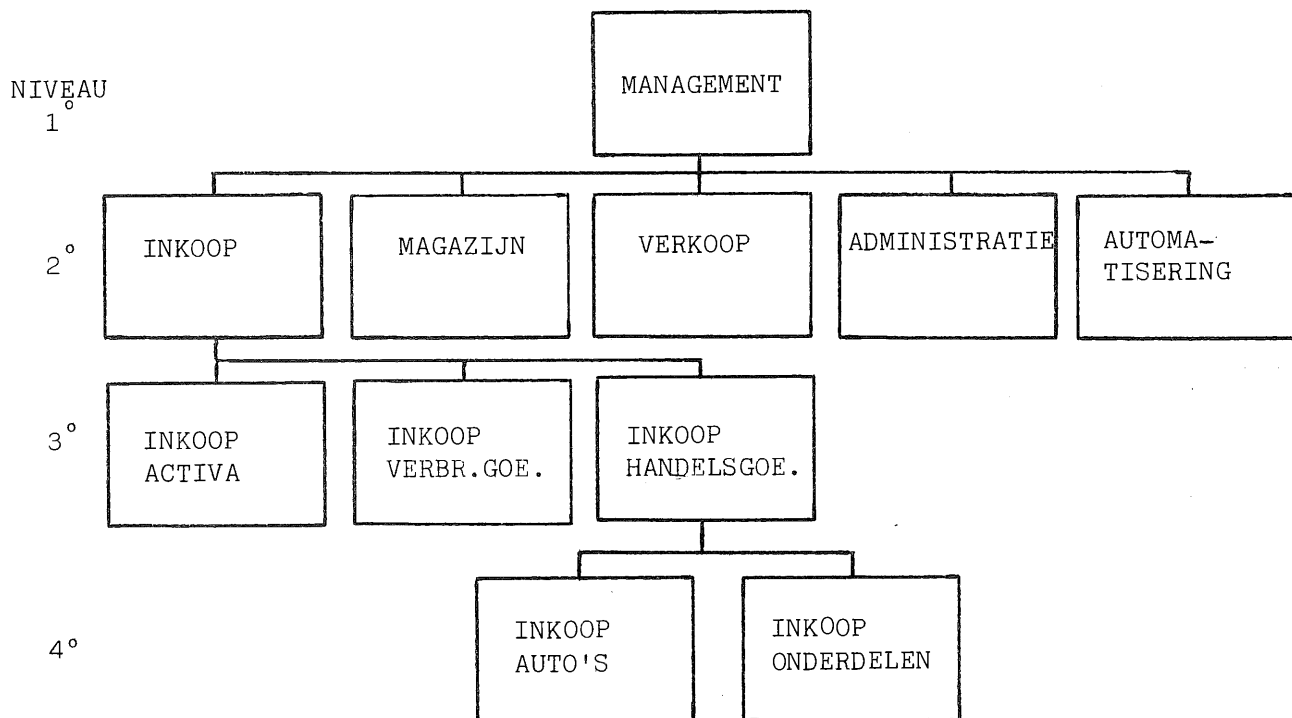


Fig. 3. Organisatie, weergegeven als samenstel van functies (voor functie INKOOP nader uitgewerkt).

In de volgende hoofdstukken zal alleen gesproken worden over functies en niet over subfuncties.

5.2 Selectie van vitale functie(s)

Om te kunnen komen tot een selectie van één of meer vitale functies dienen eerst de bedrijfsactiviteiten te worden geïnventariseerd en vastgelegd. Meestal is een vastlegging beschikbaar in de vorm van een organisatieschema. Van geval tot geval zal onderzocht moeten worden of hetgeen door het organisatieschema wordt aangegeven nadere detailering vereist.

Uitgaande van de vastlegging is selectie slechts nodig van die functie(s) die van vitaal belang is (zijn) voor de continuïteit van de bedrijfsactiviteiten.

(verkoopfunctie cash & carry bedrijf
 reserveringsfunctie bij luchtvaartonderneming
 inkoopfunctie brandstof luchtvaartonderneming, etc.)

5.3 Functie en automatisering

Zijn één of meerdere functies geselecteerd dan dient vastgesteld te worden of bij uitvoering van die functie(s) gebruik wordt gemaakt van de automatisering of niet. Indien van automatisering geen gebruik wordt gemaakt kan het onderzoek als beëindigd worden beschouwd waarbij de conclusie is dat geen continuïteitsafhankelijkheid bestaat met betrekking tot de automatisering. Bij automatisering zal vastgesteld dienen te worden in welke mate de functie ervan afhankelijk is. Hierop wordt in paragraaf 5.4 nader ingegaan. Eerst een ander aspect.

In de praktijk, wanneer een functie gebruik gaat maken van automatisering, wordt gauw gezegd dat een geautomatiseerd systeem voor die functie wordt ontwikkeld. Anders gezegd, de functie gaat gebruik maken van geautomatiseerd systeem "X". Meestal krijgt dat "systeem" de naam van die functie waarmee de relatie met de bedrijfsactiviteit is gelegd. Volgens de schrijver is deze praktijk er de oorzaak van, dat de automatisering als "iets bijzonders" wordt gezien. Ze dient echter te worden gezien als een wezenlijk onderdeel van die functie zelf. Indien het geautomatiseerde deel wordt uitgeschakeld is de functie als zodanig uitgeschakeld.

Het is daarom beter te spreken van een geautomatiseerd deel van een informatiesysteem. In figuur 4 is de relatie functie-automatisering weergegeven, zoals deze dient te worden gezien. In de benaderingswijze is als uitgangspunt genomen de totale functie. Automatisering wordt op deze manier gezien als "middel tot".

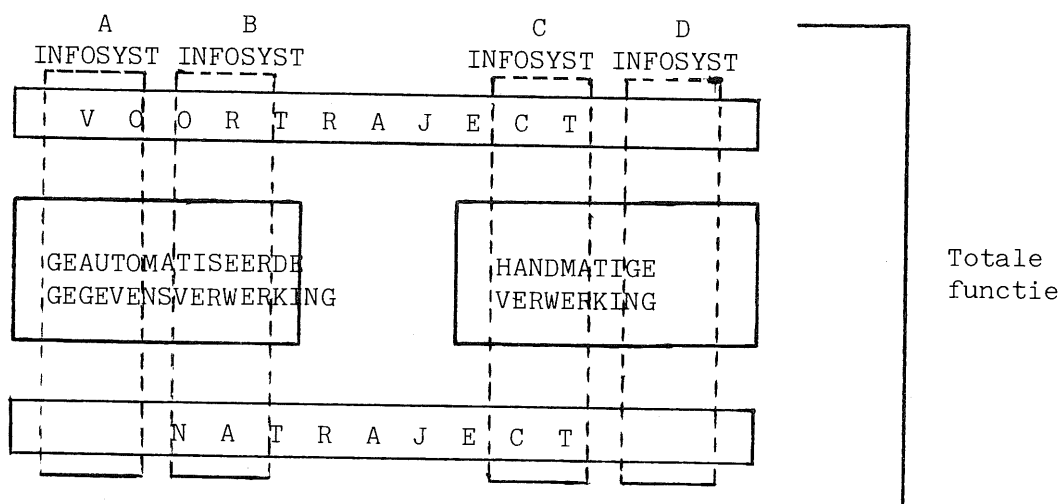


Fig. 4. Relatie functie VS automatisering.

5.4 De "cag"

Als een vitale functie gebruik maakt van de automatisering, houdt dat niet meteen in dat er sprake is van continuïteitsafhankelijkheid met betrekking tot de automatisering. Die afhankelijkheid varieert met de Maximaal Acceptabele Vertraging (MAV) van de totale functie. Hoe kleiner de MAV, hoe groter de cag.

De MAV is een termijn, die het management in geval van storing maximaal acceptabel vindt boven de normale doorlooptijd van de activiteiten binnen een vitale functie, omdat anders de continuïteit van de onderneming gevaar kan lopen. In bepaalde gevallen zal deze MAV "o" kunnen zijn, waarbij dus ook geen uitval van de automatiseringsfunctie is toegestaan.

5.5 De MAV

De bepaling van de grootte van de MAV is zuiver een beleidskwestie. Uitgaande van een aantal beleidsbepalende factoren (fig. 6) dient het management aan te geven, welke vertraging een vitale functie mag hebben. Bij een normale doorlooptijd van 2 weken en een MAV van 1 week is de maximaal toegestane doorlooptijd van de activiteiten binnen de functie 3 weken. Dit kan inhouden, dat het geautomatiseerd deel van het systeem maximaal 1 dag buiten werking mag zijn.

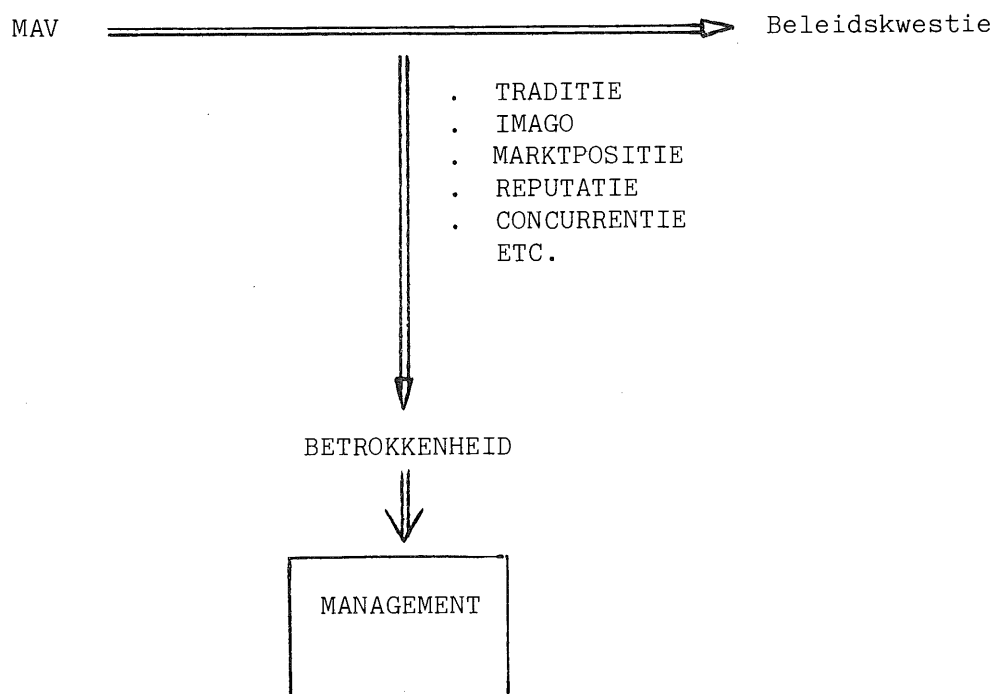


Fig. 6. Relatie MAV, beleidsbepalende factoren & management.

Geen algemeen antwoord is te geven op de vraag of die afhankelijkheid bestaat bij een MAV van één dag of twee weken. Dat zal van geval tot geval verschillen. In deze fase zal het antwoord gegeven moeten worden op de vraag of er sprake is van continuïteitsafhankelijkheid met betrekking tot de automatisering. Bij een negatieve conclusie kan het onderzoek als beëindigd worden beschouwd. Het navolgende gaat in op wat er bij twijfel of geconstateerde afhankelijkheid dient te worden gedaan.

5.6 De MUF

Onder de Maximale Uitvaltijd van de vitale Functie wordt verstaan de uitvaltijd van de functie, die zich - gezien de werkelijke situatie - maximaal zal voordoen bij storingen. In tegenstelling tot de MAV, die een zaak van het management is, is de MUF een operationele zaak. De MUF zal in hoofdstuk 6 nader worden uitgewerkt waarbij de vraag - bij twijfel of geconstateerde continuïteitsafhankelijkheid - dient te worden beantwoord (fig. 7).

"Zijn er zodanige beveiligingsmaatregelen getroffen dat de MAV \geq Maximale Uitvaltijd van de vitale Functie."

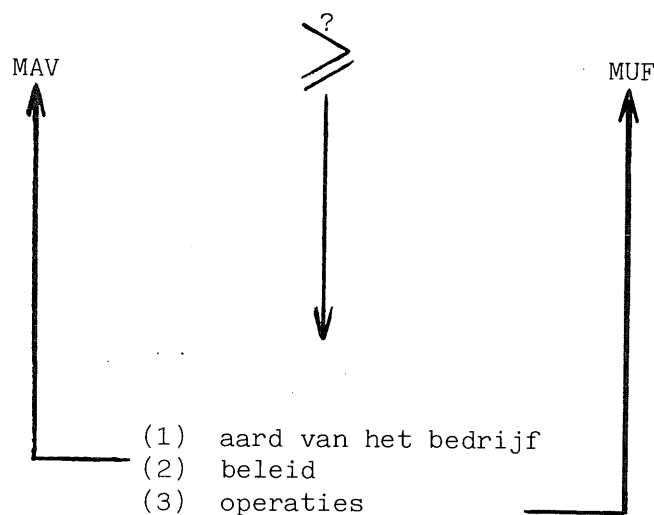


Fig. 7. Uitwerking CAG.

6.0 Vaststelling MUF

6.1 Bepalende elementen

Tot nu toe is gesproken over de vergelijking $MAV \geq MUF$ met betrekking tot de totale functie. Deze vergelijking geldt eveneens voor de drie componenten van de totale functie namelijk voortraject, het geautomatiseerd deel en het natraject. In figuur 8 is een en ander schematisch weergegeven.

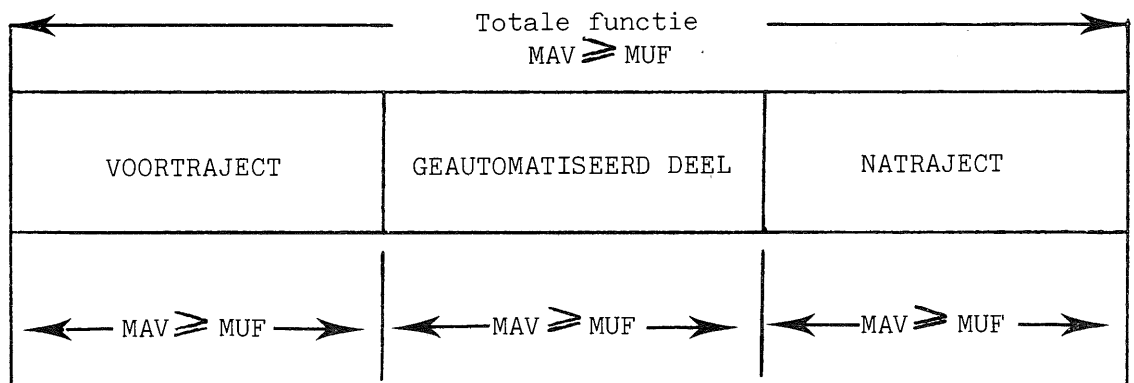


Fig. 8. Relatie componenten ten opzichte van totale functie met betrekking tot MAV/MUF.

Ter wille van de duidelijkheid wordt verdergegaan met de vergelijking $MAV \geq MUF$ voor het geautomatiseerde deel van de functie. De werkwijze met betrekking tot het voor- en natraject zijn identiek. Bij nadere uitwerking blijkt dat het gebied dat bepalend is voor de MUF, uiteenvalt in de deelgebieden omgeving, apparatuur, programma-tuur, mensen en organisatie. In de volgende paragrafen wordt zeer globaal aangegeven, wat met de deelgebieden wordt bedoeld en op welke aspecten de aandacht moet worden gericht om een MUF te kunnen bepalen (nadere uitwerking van deze deelgebieden vindt thans plaats binnen de werkgroep "Beleidsplan en risico-analyse" van het NGI).

6.2 De omgeving

Hiermede wordt de omgeving - gebouw, terrein - bedoeld waar de geautomatiseerde gegevensverwerking plaatsvindt. Een oordeel moet worden gevormd inzake de genomen beveiligingsmaatregelen en, gezien die maatregelen, wat de werkelijke uitvaltijd maximaal bij storing zal zijn. Door inventarisatie van de genomen maatregelen en gesprekken met functionarissen kan daarin inzicht worden verkregen.

') Van de invloed die de deelgebieden op elkaar hebben is vooralsnog geabstraheerd. Onderkend wordt dat die invloed er is.

6.3 De apparatuur

Hieronder wordt verstaan alle apparatuur, waarvan het geautomatiseerde deel van het systeem gebruik maakt. Niet elk toepassingsprogramma maakt altijd gebruik van alle apparatuur. Het is daarom van belang slechts die apparatuur in het onderzoek te betrekken, die nodig zijn om de vraag "is $MAV \geq MUF$ " te kunnen beantwoorden. Bepaal daarom per vitale functie welke geautomatiseerde delen van informatiesystemen gebruik maken van welke hardware componenten.

Die hardware componenten worden dan: kritische componenten genoemd (fig. 11).

Bedrijfs- functie Hardware	INFO-SYST. A		INFO-SYST. B		
	Progr.	Progr.	Progr.	Progr.	Progr.
	I	II	I	II	III
CPU	X ¹⁾	X	X	X	X
Terminal	X	-	X	-	X
Tape-unit	X	X	-	-	X
Schijf-unit	X	-	X	X	X
Printer	-	X	-	-	
Modem	X	-	X	-	X

¹⁾ dit zijn kritische componenten.

Fig. 11. Bepaling van kritische componenten.

Inzicht moet worden verkregen in het feit welke kritische component de maximale uitvaltijd veroorzaakt gezien de getroffen beveiligingsmaatregelen. Storingsoverzichten kunnen inzicht verschaffen omtrent de aard en de duur van de storingen veroorzaakt door kritische componenten. (In de praktijk plegen deze overzichten nogal eens te ontbreken.)

6.4 De programmatuur

Ook hier kunnen storingsoverzichten inzicht verschaffen in de aard en de duur van de storingen veroorzaakt door de software. De klemtoon inzake de beveiliging dient te liggen op procedures en maatregelen met betrekking tot reconstructie. Als een aantal herstelbepalende factoren kunnen worden genoemd:

- mate van structurering van programma's;
- vermijden van complexiteit van programma's;
- aanwezigheid van documentatie;
- aanwezigheid van meerdere versies, etc.

6.5 Mensen en Organisatie

Hiermede worden de mensen en de organisatie bedoeld - op de afdeling Automatisering - waarvan de continuïteit van het geautomatiseerde deel van het systeem van de functie afhankelijk is. Zijn maatregelen genomen om de uitvaltijd zo klein mogelijk te houden. Om daarin inzicht te verkrijgen kan nagegaan worden of onder andere aan de volgende aspecten aandacht is besteed:

- spreiding van kennis (niet bij één persoon);
- screening personeel;
- overdracht van kennis bij vertrek;
- vervanging bij incompetentie.

6.6 Conclusie

Is eenmaal de MAV gegeven en de MUF bepaald dan kan de conclusie worden getrokken of de beveiligingsmaatregelen voldoende zijn om aan de door het management gestelde eisen te kunnen voldoen.

Blijkt dat de MUF \Rightarrow MAV dan staan er twee wegen open:

- aanvullende beveiligingsmaatregelen moeten worden genomen opdat de MUF kleiner wordt dan de MAV;
- het management besluit geen aanvullende beveiligingsmaatregelen te nemen en een MAV te accepteren die gelijk is aan de MUF.

Het besluit om een grotere MAV te accepteren in plaats van het nemen van aanvullende beveiligingsmaatregelen zal meestal berusten op kostenoverwegingen.

7.0 Een voorbeeld

Van een onderneming is gebleken dat de functie Verkoop vitaal is voor de continuïteit van de onderneming. Zij maakt gebruik van de informatiesystemen A en B die gedeeltelijk zijn geautomatiseerd.

A bestaat uit 2 programma's;

B bestaat uit 3 programma's.

Het management accepteert een MAV van 5 dagen voor het geautomatiseerd gedeelte van het systeem waar de functie gebruik van maakt. Bij de bepaling van de MUF is het volgende gebleken.

Progr.	Hersteltijd(dagen)	Aantal storingen/maand
A {	1	2
	2	3
B {	3	7
	4	1
	5	½
		4
		1
		0
		10
		20

Het management besluit uit kostenoverweging een MUF van 7 dagen (progr. 3) te accepteren omdat de storingsfrequentie 0 was.

8.0 CAG en systeemontwikkeling

In het voorgaande is gesproken over reeds bestaande geautomatiseerde delen van systemen. Het verdient aanbeveling reeds bij de ontwikkeling rekening te houden met de beveiligingseisen die er uit oogpunt van een functie aan moeten worden gesteld. In de praktijk wordt naar de mening van de schrijver hieraan te weinig systematisch aandacht besteed. De aspecten, die bij de bepaling van de MUF een rol spelen, dienen bij de ontwikkeling van geautomatiseerde informatiesystemen te worden betrokken.

9.0 Verzekeringen

Een aspect dat niet met name genoemd is bij de bepaling en de uitwerking van de CAG is verzekeringen.

In het kader van de beveiligingsproblematiek is het zaak om risico's af te dekken of zo veel mogelijk te beperken. Een middel om risico's te beperken is zich er tegen te verzekeren. Een punt dat hierbij van belang is, dat niet elk risico bij automatisering te verzekeren is (kleine letters), waarvan akte.

10.0 Slotopmerkingen

Getracht is in hoofdlijnen een benaderingswijze te introduceren die de problematiek van de beveiliging inzake automatisering benadert vanuit de continuïteitsoptiek van de onderneming. Te zijner tijd zal in Compact gedetailleerder op een aantal facetten van deze benaderingswijze worden ingegaan.

Programmaconversie - systeemontwikkeling met handicap.
=====

door J.C.P.M. Vermeeren

1. Inleiding

In een van de eerste nummers van "COMPACT" (2e jaargang, nummer 1, voorjaar 1975) verscheen een artikel over het onderwerp bestandsconversie. In de inleiding van dat artikel werd reeds aangegeven dat de bestandsconversie slechts één van de elementen van de conversieproblematiek is. Andere elementen zijn de omzetting van programmatuur, verandering van installatie, operating systeem, wijziging van procedures en mogelijk zelfs wijziging van de organisatiestructuur. In dit artikel zal worden ingegaan op het element omzetting van toepassingsprogramma's. Met "omzetting" wordt bedoeld een aanpassing aan de nieuwe situatie, waarbij geen wijziging in resultaat of doelstelling wordt beoogd. Aan de overige elementen van het conversieproces wordt in dit artikel slechts aandacht besteed voor zover dat voor het onderwerp van belang is.

2. Terreinverkenning en begrippen

In de literatuur wordt onder programmaconversie verstaan de omzetting van de verwerkingsprogramma's nodig geworden door :

- overgang naar een andere computer;
- overgang op een andere programmataal;
- overgang op een ander besturingssysteem;
- invoering van of overgang naar gegevensbanken of verandering van type bestandsstructuur;

Deze definitie geeft een goed beeld van de reden doch weinig inzicht in de aard van het proces.

Men bedenke wel, dat de omvang en het belang van de programmaconversie in ieder individueel geval verschillend kunnen zijn. De invloed van de in de definitie genoemde veranderingen beweegt zich tussen twee uitersten :

1. De verwerkingsprogramma's (systemen) behoeven geen aanpassing. Volstaan kan worden met de constatering dat dat ook werkelijk het geval is.
2. De vereiste verandering zal van dien aard zijn, dat in feite geheel nieuwe programma's moeten worden geschreven.

Bij de overgang naar een nieuwe computer staan in wezen twee wegen open :

1. De bestaande systemen worden in beginsel ongewijzigd gehandhaafd. Ze dienen alleen te worden aangepast aan de nieuwe omgeving.
2. De bestaande systemen worden herschreven om op die manier de voordelen van de nieuwe omgeving ten volle te benutten.

Onder omgeving dienen alle middelen te worden begrepen die op het functioneren van het programma inwerken. Een aantal belangrijke omgevingscomponenten zijn genoemd in de eerste alinea van deze paragraaf.

De programmaconversie is een formeel proces. Aangezien het programma vóór en ná conversie dezelfde functies dient te verrichten, zijn materiële wijzigingen niet gewenst. Er kunnen zich gevallen voordoen waarbij de invloed van de omgeving zo groot is dat in de praktijk deze formele benadering niet haalbaar is. Voorts kan de nieuwe omgeving zodanige voordelen bieden dat op economische gronden besloten wordt, zonder de programmaconversie als tussenfase, direct nieuwe, verbeterde, toepassingen te ontwikkelen. Anders dan bij bestandsconversie is er bij programmaconversie derhalve een keuzemogelijkheid.

Nieuwe systemen die de nieuwe situatie op het lijf geschreven zijn, zullen uiteindelijk bijna steeds de beste keuze zijn. Nadeel van deze benadering is de vaak veel langere doorlooptijd van het aanpassingsproces. Gedurende die periode moet dan, in geval van overgang op een nieuwe computer, twee computersystemen operationeel worden gehouden. In de afweging van beide alternatieven dient ook de mate, waarin de systemen nog aan door gebruikers te stellen eisen voldoen, te worden betrokken.

In dit artikel zal onder programmaconversie worden verstaan de omzetting van verwerkingsprogramma's (voortvloeiend uit wijzigingen in de apparatuur en/of besturingsprogrammatuur in de ruimste zin) zonder dat daarbij het oogmerk bestaat die programma's materieel te wijzigingen. In deze definitie wordt onder verwerkingsprogramma mede begrepen de voor de uitvoering ervan benodigde besturingsopdrachten (Job Control).

3. Oorzaak en gevolg

In de eerste alinea van de vorige paragraaf is gebleken dat de behoefte aan programmaconversie verschillende oorzaken kan hebben. De invloed daarvan op de omvang en de reikwijdte van de conversie hangt van vele factoren af. Zonder volledigheid na te streven worden er hier onder een aantal genoemd.

1. de taakverdeling tussen het programma en het besturingssysteem kan veranderen.
2. de besturingstaal is doorgaans machine- en besturingssysteem gebonden.
3. de hogere programmeertalen zijn weliswaar gebaseerd op standaardconventies, doch worden door de leveranciers van "eigen" extra's voorzien.
4. de wijze van datarepresentatie en de mogelijke dataformaten zijn doorgaans leverancier- en machinegebonden.

5. de communicatie tussen de centraleverwerkingseenheid en de randapparatuur is machinegebonden.
6. de verandering van computersysteem brengt vaak veranderingen mede ten aanzien van het type gegevensdrager waarop de bestanden worden bewaard.
7. De opslagstructuur van de gegevensverzamelingen kan niet onbelangrijk wijzigen.

Op de problematiek rond de verandering van programmeertaal en/of besturingssysteem zal niet afzonderlijk worden ingegaan, omdat deze als onderdeel van de overgang op een nieuwe computer reeds aan de orde komen.

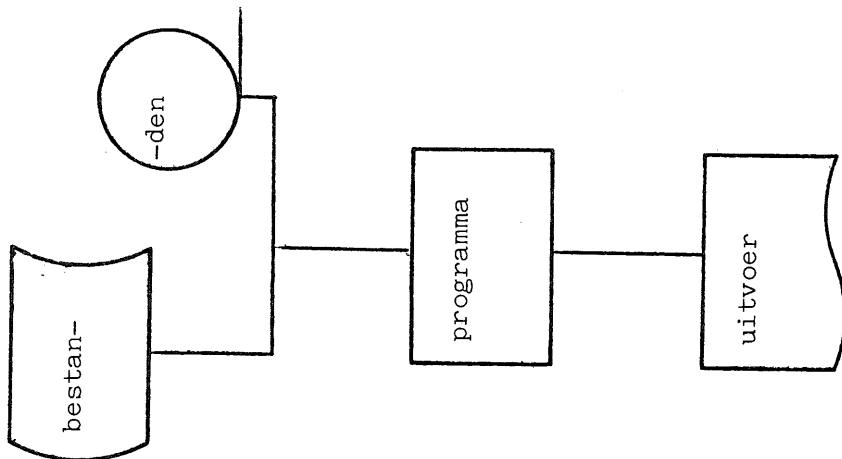
4. Controle-overwegingen

Uit controle overwegingen dient aan de programmaconversie als eis te worden gesteld, dat in de nieuwe situatie -gegeven correct geconverteerde bestanden- met de omgezette programma's dezelfde resultaten worden verkregen als voor de conversie het geval was. Hierbij dient in acht te worden genomen dat ook de controle-middelen zowel ten behoeve van de gebruikers als ten behoeve van de input/output controle binnen het computercentrum, ter vaststelling van volledigheid, juistheid en tijdigheid van de gegevensverwerking ongewijzigd in takt dienen te blijven. Wellicht ten overvloede wordt nog beklemtoond, dat de controleerbaarheid van de informatieverwerking ook gedurende het conversieproces gewaarborgd dient te blijven.

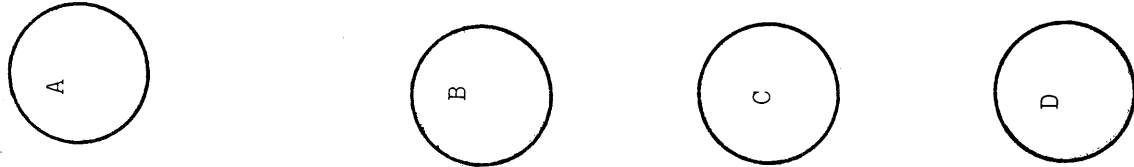
Controlegebieden : (zie schema 1)

- A. Het controlegebied A omvat het geheel van controlemaatregelen met betrekking tot de bestandsconversie. Voor de uitwerking daarvan wordt verwezen naar het in de inleiding genoemde artikel. De controle is zowel gericht op volledigheid als op juistheid.
- B. Het controlegebied B is gericht op een positieve controle van het feitelijke omzettingsproces. Men denke hierbij aan de constatering dat ieder programma inderdaad is omgezet en dat waarneembaar is dat zich daarbij geen materiële veranderingen voordeden.
- C. Het controlegebied C richt zich op de resultaten van het programma. De controle wordt verricht door de vergelijking van de uitvoer vóór en ná de conversie. In principe mogen hierbij géén verschillen blijken. De controle is negatief gericht en heeft de programmafunctie als object.
- D. Het controlegebied D richt zich op de organisatie van het conversieproces. Een belangrijke maatregel van functiescheiding uit controletechnisch oogpunt is die tussen de bestands- en de programmaconversiefuncties.

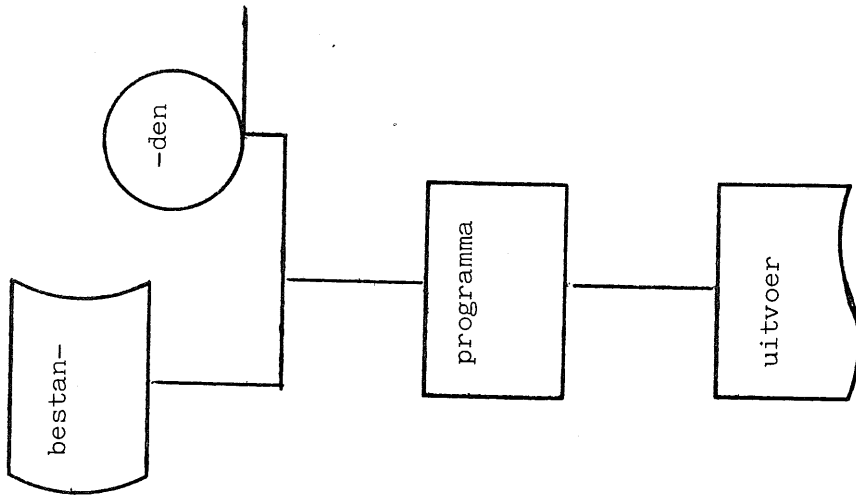
oorspronkelijk



controle gebieden



geconverteerde



De hiervoor weergegeven controle-aanpak is gebaseerd op het uitgangspunt, dat indien de volledigheid en de juistheid van de geconverteerde bestanden eenmaal is vastgesteld, door vergelijking van de resultaten kan worden gecontroleerd, dat het programma door de conversie niet inhoudelijk is gewijzigd. Een strikte functiescheiding tussen bestands- en programmaconversie kan dan niet worden gemist. Aangevuld met een positieve controle op de programmaconversiewerkzaamheden wordt een sluitend complex van controlemaatregelen verkregen.

Het is de vraag of ingeval van conversie van ingewikkelde bestanden, waaronder data base, deze controlefilosofie gehandhaafd kan blijven. De vaststelling van de volledigheid en juistheid van de bestandsconversie kan dan zodanig worden bemoeilijkt, dat daar een onvoldoende mate van zekerheid wordt verkregen om te dienen als peiler voor de negatieve controle op de programmaconversie. Een oplossing voor die problematiek wordt in deze bijdrage niet gegeven. Bedacht dient wel te worden, dat in dit geval doorgaans ook geen sprake zal zijn van een programmaconversie als gedefiniëerd in paragraaf 2.

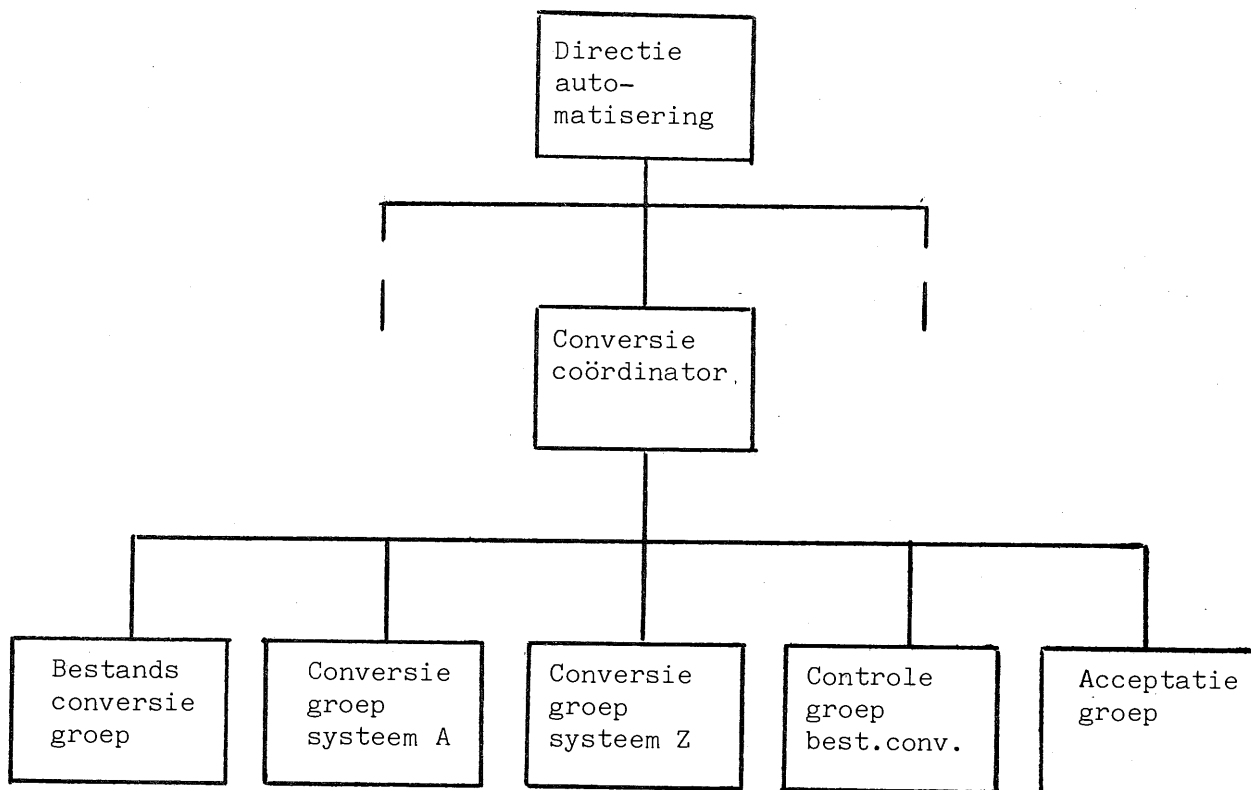
Tot nu toe is in deze paragraaf gesproken over programma's. De geschetste controlemethodiek kan echter op verschillende nivo's worden gebruikt. Van delen van programma's tot en met toepassingssystemen.

De controlewerkzaamheden in het kader van de programmaconversie worden door diverse functies verricht. Op het nivo van programma's of programmadelen zal zelfcontrole van de conversieprogrammeur of van de conversiegroep een belangrijke plaats innemen. Op programma- en subsysteemnivo zal een verbijzonderd interne controleorgaan en/of de acceptatiegroep een belangrijke functie vervullen. In de acceptatiegroep kan de accountant (intern en/of extern) participeren. Ook de gebruikers van het subsysteem kunnen op dit punt een goede bijdrage leveren. Op systeemniveau zal de participatie van de gebruikers alsmede van in- en externe accountant groot moeten zijn. In ieder geval zal de externe accountant zulk een inbreng dienen te realiseren dat hij uit eigen waarneming tot het oordeel komt, dat aan de genoemde controle eis wordt voldaan.

Het zal duidelijk zijn dat een goede documentatie van het conversieproces voor de accountantscontrole, en niet alleen daarvoor, een belangrijk hulpmiddel is.

5. Organisatie

In het navolgend schema wordt een mogelijke uitwerking van de structuur van de conversie-organisatie weergegeven. Aan de hand daarvan worden de onderscheiden functies besproken.



- Directie automatisering :

Deze geeft algemene leiding aan het gehele overgangsproces waarvan de conversie slechts een aspect is. De installatie van de nieuwe computer en de ontwikkeling van systemen voor de nieuwe computer zijn twee belangrijke andere activiteiten.

- Conversie coördinator :

Dat het conversieproces een veel omvattende klus is, dat bovendien een veelheid van vooraf niet te identificeren problemen met zich brengt leidt tot de noodzaak om hier een krachtige en ervaren coördinator te benoemen die niet belast is met de feitelijke uitvoering van het proces.

- Bestandsconversiegroep :

De bestandsconversiegroep draagt zorg voor het juist en volledig omzetten van de bestanden. Hierbij wordt rekening gehouden met algemene afspraken en de eisen van de programmaconversiegroepen. Hoewel het omzetten van de bestanden een geheel andere deskundigheid vereist als de programmaconversie kan worden gekozen voor een conversiegroep per systeem waarin zowel de bestandsconversie als de programmaconversie wordt verricht. Met betrekking tot de eis van functiescheiding dient dit uit controle-oogpunt als een verzwakking te worden gezien.

- Conversiegroep Systeem A....Z :

In deze groepen worden de werkzaamheden met betrekking tot de programmaconversie verricht. In de volgende paragraaf zal op de werkwijze nader worden ingegaan.

- Controle op geconverteerde bestanden :

Alle geconverteerde bestanden worden door een afzonderlijke controlegroep op juistheid en volledigheid onderzocht. Ook hier doet zich een mogelijk alternatief voor. Indien afzonderlijk conversiegroepen voor bestands- en programmaconversie bestaan, dan kan het zinvol zijn de controle op de bestandsconversie door de programmaconversiegroep te doen verrichten. De hierna te bespreken acceptatiegroep houdt dan toezicht op een toereikende uitvoering van die controletaak.

- Acceptatiegroep :

Deze groep voert met de geconverteerde programmatuur en de gecontroleerde bestanden een systeemtest uit waarbij de resultaten van het oude en het nieuwe systeem worden vergeleken. In deze groep zijn in ieder geval de produktieafdeling en systeemprogrammering vertegenwoordigd.

De aandacht wordt er op gevestigd dat noch de gebruiker noch de accountant in het voorgaande is genoemd.

Soms wordt betoogd, dat aangezien er geen inhoudelijke wijziging plaatsvindt, de gebruikersinbreng achterwege kan blijven. Dit argument is betrekkelijk omdat er impliciet van wordt uitgegaan dat dat een voornemen (het programma niet te wijzigen) als een vaststaand resultaat wordt aangenomen. De gebruiker van het systeem moet als uiteindelijk verantwoordelijke voor de geautomatiseerde delen van zijn administratie nagaan of die belofte gestand is gedaan. Doorgaans zal blijken dat ondanks alles toch geringe wijzigingen onontkoombaar blijken.

Daarom moet, bij voorkeur, de leider van de conversiegroep de gebruiker betrekken bij de beslissing omtrent ongewilde doch noodzakelijke veranderingen. Daarenboven zal de acceptatiegroep een systeem niet voor produktie mogen vrijgeven voordat de gebruikers zich daarmee schriftelijk akkoord hebben verklaard. De accountantsinbreng kan ten aanzien van de programmaconversie slechts ten dele worden geëffectueerd door deelname in de acceptatiegroep. De waarneming van zowel de organisatie als voortgang en uitvoering van de conversietaken vereisen een meer onafhankelijk gezichtspunt, dandoor participatie in de acceptatiegroep die een zeker collisiegevaar met zich mee brengt.

6. Het conversieproces in stappen :

Het conversieproces is een iteratief proces. Doorgaans zal pas na een aantal pogingen een schoon en correct programma worden verkregen. Het proces kent ook een aantal fasen die min of meer na elkaar moeten worden doorlopen.

6.1. Voorbereidingsfase algemeen :

Goede voorbereiding is het halve werk. Twee dingen dienen te worden bepaald :

- wat veranderd moet worden;
- hoe zullen die veranderingen worden doorgevoerd.

Uit de vergelijking van de eisen die de oude en de nieuwe omgeving stellen aan de toepassingsystemen kan doorgaans een set standaards worden afgeleid.

In COBOL termen zullen deze standaardwijzigingen in de eerste plaats te vinden zijn in de environment division van het programma. Ook de data division zal echter wijzigen door veranderde eisen met betrekking tot data-formaten. Voorts kan de oplossing die voor de bestandsconversie wordt gekozen voor het programma veranderings-behoeften doen ontstaan.

Ook de procedure division zal veranderen. Bijvoorbeeld door andere conventies voor het open en sluiten van bestanden, het werken met tabellen en dergelijke. Bovendien zullen, waar gebruik wordt gemaakt van door de leveranciers aan programmeertalen toegevoegde extra's, deze vertaald moeten worden naar de nieuwe situatie.

Veel van deze veranderingen zijn voorzienbaar. Daarvoor kunnen standaardoplossingen worden afgesproken en vastgelegd in een conventie. Het zal een illusie zijn en blijven om in normale gevallen alle wijzigingen in die wijzigingsconventie te vatten. Juist deze omstandigheid maakt het programmaconversieproces zo moeilijk controleerbaar.

6.2. Voorbereiding per systeem :

Per systeem (of subsysteem) zal de conversieprocedure ter hand worden genomen waarbij de bestands- en programmaconversie, weliswaar onafhankelijk van elkaar, goed gecoördineerd dienen te verlopen. De in paragraaf 6.1. genoemde wijzigingsconventie zal mede als uitgangspunt dienen voor afspraken ten aanzien van :

- bestands en record lay-outs (sorteervolgorde, blokkingsfactoren, recordlengten);
- testsets (de bestaande testsets dienen te worden aangepast aan de situatie na omzetting);
- voortgangsplanning en communicatie.

6.3. Het omzettingsproces :

Per programma wordt nu het omzettingsproces ter hand genomen. Dit proces richt zich niet alleen op het programma zelf maar ook op de set besturingsopdrachten nodig om het programma uit te voeren.

Bij het omzettingswerk kan gebruik gemaakt worden van hulpmiddelen zoals :

- programma's die, bestuurd door parameters, de te standaardiseren omzettingen realiseren;
- utilities en compileerprogramma's;
- vergelijkingsprogramma's gericht op de uitvoer.

Doorgaans zal deze programmatuur het conversiewerk niet zonder menselijk ingrijpen tot een goed einde kunnen brengen. Het belang ervan is daarom uit controle-oogpunt beperkt.

Nadat de feitelijke omzetting heeft plaatsgevonden worden programma- en besturingsopdrachten onderzocht op taalfouten met behulp van het compileerprogramma van de nieuwe machine en de Job Control Syntax Checker. Als beide "schoon" zijn dan kan het programma worden getest. De programmatest heeft als doel vast te stellen, dat het geconverteerde programma dezelfde functies vervult en dus tot dezelfde resultaten leidt als de oorspronkelijke versie. Dit lijkt eenvoudiger dan het is. Idealiter dienen n.l. twee testsets te worden opgesteld, (voor beide programma-versies één) waarbij aan twee eisen moet worden voldaan :

- de resultaten moeten identiek zijn ;
- de sets moeten volledig zijn dat wil zeggen ze moeten rekening houden met alle verschillen die er tussen de beide versies en hun respectievelijke omgeving bestaan.

Zeker in complexere situaties is dit niet realiseerbaar. In de praktijk kan het een goede oplossing zijn met drie testsets te werken.

1. één testset gericht op de eigen karakteristiek van het nieuwe programma;
2. twee testsets gericht op de vergelijking van de resultaten van het programma vóór en ná de conversie. Hieruit kunnen dan tevens de eventueel benodigde bijstellingen van de vergelijkings-programmatuur ten behoeve van de parallel-productie worden afgeleid.

In deze aanpak zijn de twee hiervoor afgeleide eisen ten aanzien van de testsets van elkaar losgemaakt.

6.4. Systeemtest :

Nadat alle programma's en bestanden zijn geconverteerd kan de systeemtest worden uitgevoerd.

Voor de controlepunten wordt verwezen naar het schema in paragraaf 4.

Bij de systeemtest kan gebruik worden gemaakt van twee elkaar aanvullende technieken :

1. testgevallen;
2. parallel- of schaduwproductie.

In beide gevallen kan voor de vergelijking van resultaten van het oorspronkelijke en geconverteerde systeem gebruik worden gemaakt van vergelijkingsprogrammatuur.

De parallelproductie dient een volledige systeemcyclus te omvatten. Dat wil zeggen dat ook het kwartaal- en jaarwerk en zg. incidentele systeemgedeelten in de test dienen te worden betrokken.

6.5. Acceptatie en invoeringsfase :

Uit de systeemtest, waarbij de gebruiker betrokken moet zijn, volgt eventueel na een correctiecyclus vanzelf de formele acceptatie van het geconverteerde systeem door gebruiker, input-/outputcontrole, en produktieafdeling.

Afhankelijk van de aard van de opdracht kan in deze fase ook een schriftelijke goedkeuring door de accountant plaatsvinden.

6.6. Nazorg-fase :

Zoals ook gebruikelijk na de invoering van nieuwe systemen volgt nu de nazorg-fase waarin eventueel achterstallig onderhoud op het systeem kan plaatsvinden. Gezien de ongewisheid die ondanks alle voorzorgen aan de conversie van programma's blijft kleven, gecombineerd met een betrekkelijke onervarenheid met het nieuwe computersysteem, dient men gedurende langere tijd alert te blijven op eventuele "missers" tijdens de conversie.

7. Onderhoud van systemen :

Een bijzonder aspect dat in verband met de conversie niet onvermeld mag blijven is de behoefte aan systeemwijzigingen voortvloeiend uit de dynamiek van het bedrijfsgebeuren.

Uit het voorgaande zal de lezer hebben opgemaakt dat het conversieproces een langdurige aangelegenheid is. Om die redenen kunnen vaak gewenste systeemwijzigingen niet tijdig worden doorgevoerd.

Van gebruikerszijde kan dan druk worden uitgeoefend om tijdens het conversieproces ook -en passant- die wijzigingen mee te nemen. Dit is niet zonder risico's. Mede door de verschillen in uitgangspunt tussen conversie- en onderhoudswerk compliceert dit de gang van zaken ontzettend. De mogelijke gevolgen voor de kwaliteit van de geconverteerde systemen laten zich raden. Bovendien kan de in deze bijdrage geschetste controle-aanpak niet meer worden gevolgd.

Gezien bovenstaande argumentatie dient een vermenging van onderhouds- en conversiewerk te worden vermeden.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

door A.H.C. Koedijk

Overzicht

In april/mei 1981 ben ik in de gelegenheid gesteld een studiereis te maken naar Noord-Amerika. Voornaamste aanleiding voor zo'n reis wordt gevormd door de Conference on Computer Audit, Control & Security, georganiseerd door Harold Weiss, een autoriteit op het AC-vakgebied en onder meer uitgever van EDPACS, de EDP Audit Control and Security Newsletter.

Weiss ondernam deze activiteit altijd in samenwerking met het Institute of Internal Auditors, doch dit jaar werd de combinatie gevormd met de EDP-Auditors Foundation. Het IIA organiseerde echter toch een eigen conferentie, de 1st Information System Controls Conference, waardoor nu in opeenvolgende weken, in dezelfde stad en zelfs in het zelfde hotel, een tweetal AC-vakconferenties waren bij te wonen.

Mijn, overigens beperkte, indruk is dat de diepgang van de IIA-conferentie iets geringer is dan die van de EDPAF-conferentie: de eerste is kennelijk wat meer gericht op "algemene" auditors, de tweede wat meer op EDP-auditors. De indruk is beperkt, doordat zo'n conferentie een groot aantal parallel sessies kent, waardoor de bezoeker niet in staat is alles waar te nemen. Dit noodgedwongen missen van sommige sessies kon dit jaar echter gedeeltelijk worden opgevangen doordat er nu een tweetal conferenties werd georganiseerd: de sprekers zijn ten dele, begrijpelijkerwijs, dezelfde. Het alsnog kunnen bijwonen van gemiste sessies enerzijds en het gegeven dat de conferenties ook een groot aantal "eigen" sprekers kennen anderzijds, rechtvaardigen naar mijn mening een bezoek aan beide evenementen.

Het programma van de door mij gemaakte reis zag er in grote lijnen als volgt uit:

- 3rd International Conference on Computer Capacity Management, georganiseerd door het Institute for Software Engineering;
- bezoeken aan Main Hurdman, zowel national office als practice offices, (KMG-partner USA);
- bezoek aan Thorne Riddell (KMG-partner Canada);
- 1st Information System Controls Conference (IIA);
- 11th Conference on Computer Audit, Control and Security (EDPAF).

Dit artikel beoogt melding te maken van enige onderwerpen die door mijzelf opvallend of interessant bevonden werden. Een volledig en tot in details uitgewerkt verslag wordt hier niet beoogd.

Het nut van een dergelijke reis schuilt ondermeer in de mogelijkheid om in vrij korte tijd over een rijke schakering aan onderwerpen iets te horen. Dit zijn niet altijd "laatste ontwikkelingen", het kunnen ook terreinen zijn waar de bezoeker nooit zo aan is toegekomen dan wel zaken die simpelweg weer eens op een rijtje worden gezet.

Voorts is het bijzonder prettig om buiten de normale werkkring en los van de werkdruk eens van gedachten te wisselen over vaktechnische onderwerpen. Niet in de laatste plaats werd ook het leggen en onderhouden van contacten, zeker ook met KMG-kantoren, door mij als nuttig ervaren.

3rd ICCCM

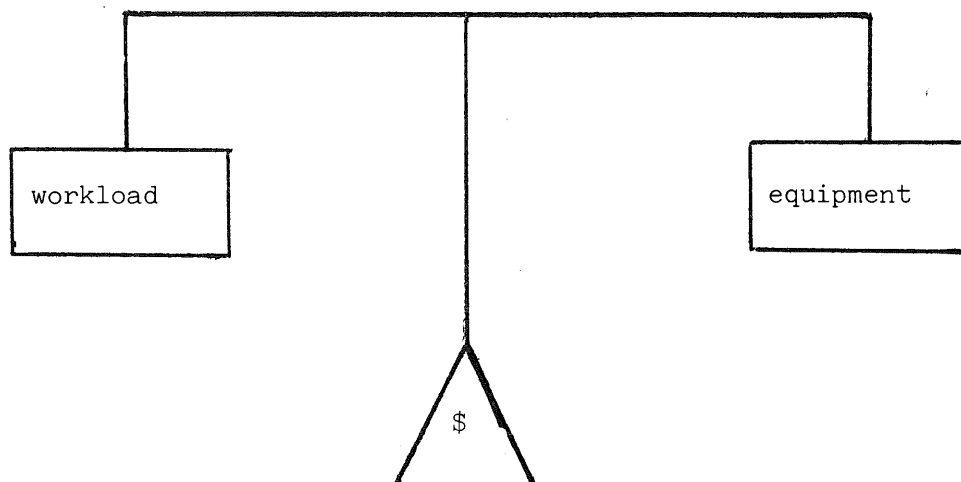
De eerste dagen van de reis brachten me in Chicago, de stad van het hoogste gebouw ter wereld (de Sears Tower, 433 meter/110 verdiepingen hoog), Al Capone en Hugh Hefner: zijn Playboy building vormde echter niet het doel van het bezoek aan deze stad. Dat werd gevormd door de computer capacity management conferentie.

Capacity management specialisten (die technische kennis vergelijkbaar met die van systeemp programmeurs moeten hebben) zijn, zo werd duidelijk, nog eenzaamroependen in de EDP-woestijn. Ze zijn derhalve enigszins vergelijkbaar met EDP-auditors. Zij beoefenen een vrij nieuw specialisme, mede gezien het feit dat zij pas aan hun derde jaarlijkse conferentie toe waren. EDP-auditors hielden dit jaar immers al hun elfde!

Capacity management is een activiteit die logisch volgt op performance measurement en performance evaluation. Het terrein grenst eigenlijk aan dat van de EDP-auditor (die trouwens de termen efficiency en effectiviteit, zij het met iets kleinere letters dan betrouwbaarheid, ook graag in zijn vaandel plaatst). Dit blijkt ook wel uit de sleutelwoorden: bij EDP-auditing zou hiervoor data-reliability kunnen worden ingevuld, bij capacity management is de term service-reliability op zijn plaats.

Capacity management laat zich dan ook als volgt definiëren:

"The process of establishing and maintaining a satisfactory balance between workload-requirements and equipment-capacity to meet service objectives at the least possible cost".



Hieruit blijkt de vooruitgang op dit terrein. Bewogen deze specialisten zich in het verleden voornamelijk op het gebied performance measurement en tuning, thans gaat het de richting uit van capaciteitsplanning.

Dit betekent niet dat performance metingen tot het verleden behoren. Het gebruik van de huidige capaciteit wordt nu echter in verband gebracht met de workload die dit gebruik veroorzaakt. Daarnaast worden, op basis van de "business plans", voorspellingen omtrent de toekomstige workload opgesteld. Daaruit wordt afgeleid wat de in de toekomst benodigde capaciteit zal zijn. Hiervoor is nodig: karakterisering en classificatie van de workload; dit vormde een belangrijk item op deze conferentie. Duidelijk werd echter, dat men hierover nog niet is uitgedacht.

Het gebruik van de computer in de accountantscontrole

Bij de bezoeken aan Amerikaanse en Canadese KMG-kantoren is dit onderwerp uitvoerig aan de orde geweest. Onder meer werd ingegaan op de voor "KKC-oud" nieuwe audit-packages CARS en EDP-AUDITOR. Deze pakketten vervangen het voor de fusie bij KKC in gebruik geweest zijnde retrieval pakket CA-EARL. Gebleken is dat in de USA en Canada het pakket CARS frequenter gebruikt wordt dan EDP-AUDITOR, doordat aldaar kennelijk van een grotere variëteit aan machines sprake is (van CARS zijn meer versies beschikbaar dan van EDP-AUDITOR) en doordat veelal niet beschikt kan worden over medewerkers met programmeurskwaliteiten, hetgeen voor het gebruik van EDP-AUDITOR als een vereiste werd vermeld. CARS zou dan ook veel geschikter zijn voor AC-accountants/niet-programmeur.

Als positieve kenmerken van CARS werden genoemd:

- de standaardmogelijkheden van het pakket voorzien in vrijwel alle accountantswensen;
- het definiëren van de specificaties voor de standaardmogelijkheden is eenvoudig en zeer snel te leren;
- zolang gebruik gemaakt wordt van de standaardmogelijkheden (en het door CARS gegenereerde COBOL-programma niet wordt uitgebreid met aanvullende routines) is kennis van COBOL niet vereist;
- ook doen zich dan geen debuggingsproblemen voor.

Bij Thorne Riddell is een formulier in ontwikkeling en op proef in gebruik ten behoeve van de aanvangsfase van de ontwikkeling van controleprogrammatuur. Het formulier heeft een inventariserend karakter en is bedoeld om ingevuld te worden door de algemeen accountant.

Vastgelegd dienen te worden:

1. Contactpersonen:
 - . algemene controle;
 - . automatiseringsafdeling cliënt.
2. Controledoelstellingen van de te automatiseren toepassing.
3. Tijdschema, inclusief deadline.
4. Omgeving:
 - . hardware configuratie mainframe (IBM 370, Univac 1100, etc.);
 - . beschikbaarheid van de installatie van de cliënt;
 - . systeemsoftware;
 - . gegevens over bestandsmedia en verwerkingwijze.
5. Gegevens over input files.
6. Gegevens over te printen reports.

Het formulier is afgestemd op de voor controleprogrammatuur geldende documentatievoorschriften.

De ervaringen met het formulier zijn redelijk. Het voornaamste probleem doet zich voor ten aanzien van technische termen, zoals OS, DOS, EBCDIC, etc.

Opvallend was, dat zowel in de USA als in Canada reeds de nodige micro's (Apple II, TRS 80 van Radio Shack) stonden opgesteld. Zij werden nog voornamelijk toegepast ten behoeve van intern-administratieve doeleinden zoals planning, urenregistratie en declaratiegegevens alsmede ter dienstverlening aan de algemene controle (het verrichten van rekenwerk door middel van standaardpakketten, die erg goedkoop zijn). Het voornaamste doel werd echter gevormd door het opdoen van ervaring met deze apparatuur. Verwacht werd, dat de micro's in de toekomst eveneens een rol zullen gaan spelen in de sfeer van computer assisted audit techniques, niet zozeer om cliëntenfiles te verwerken na transport hiervan, dan wel als terminal in een data communicatie netwerk.

De uitvoering van controleprogrammatuur geschiedt in de USA, net als in Nederland bij KKC, in principe op één van de volgende locaties:

- . computer van de cliënt;
- . eigen computer indien aanwezig op het practice office;
- . servicebureau.

In het laatste geval geschiedt dit veelal (begrijpelijk gezien de grote afstanden) via time sharing services. Interessant is, dat in dat geval soms tegen betaling toegang kan worden verkregen tot programmabibliotheken van derden, ook van andere accountantskantoren. Coopers & Lybrand is hiervoor bijvoorbeeld in de markt.

Ook interessant waren twee aspecten met betrekking tot computer assisted audit techniques in het opleidingspakket van onze Amerikaanse KMG-partner. Het ene aspect is dat de tweede week van de zogenaamde advanced course (gericht op de toepassing van CARS) ook bijgewoond dient te worden door de vennoot van de betrokken cursist met als doel dat ook deze de mogelijkheden en de mogelijke efficiency en effectiviteit van het gebruik van de computer in de controle onderkent. Hiermede wordt een hoger nut van de opleiding beoogd. Het andere aspect is, dat de door de cursist te ontwikkelen toepassing niet een cursus-casus is doch een in zijn praktijk voor een bepaalde cliënt daadwerkelijk te gebruiken toepassing. Hierdoor is het zelfs denkbaar dat de cursus naast de opleidingskosten ook een zekere opbrengst kent.

Als slot kan nog worden opgemerkt, dat de controleprogrammatuur in de USA en Canada een zeer groot deel van de AC-activiteiten vormt. De indruk bestaat, dat in verhouding tot systeem- en automatiseringsorganisatiebeoordeling de activiteiten met betrekking tot controleprogrammatuur een grotere plaats innemen dan bij KKC in Nederland het geval is.

Vermoedelijk kan de oorzaak gezocht worden in de omstandigheid, dat men in Noord-Amerika frequenter met "small shops" te maken heeft dan wij dat hier (nog) gewend zijn. Enigszins zwart-wit weergegeven komt de redenering dan hierop neer: uitgebreide onderzoeken zullen tot een conclusie leiden die ook vooraf al voorspelbaar is, namelijk dat de automatiseringsorganisatie onvoldoende waarborgen terzake van de interne controle zal bieden; derhalve zal het accent komen te liggen op "substantive testing", waarbij controleprogrammatuur efficiënt en effectief zal zijn, zodat beter maar meteen met de ontwikkeling hiervan kan worden begonnen.

Systeemonderzoek

Met name in Canada is hier tijdens de studiereis de nodige aandacht aan besteed. Het voornaamste verschil met Nederland werd gevonden in de rapportering. In Canada wordt gerapporteerd in de vorm van een managementletter, waarin eigenlijk alleen de "weaknesses" worden vermeld. Een overall oordeel of verklaring, een zekere aan een oordeel ten grondslag liggende documentatie (globale beschrijving van het systeem, beschrijving van het hierop geprojecteerde systeem van interne controle), begrenzing van het onderzoek worden veelal niet in de rapportering teruggevonden. Met name niet, indien het een onderzoek "in het kader van de normale jaarrekeningcontrole" betrof, hetgeen uiteraard meestal het geval is.

Uit bestudering van de dossiers kan worden afgeleid dat sprake is van:

- . in de meeste gevallen: een goede samenwerking tussen AC-medewerkers en leden uit de algemene controlegroep bij het systeemonderzoek (geautomatiseerde deel en de administratieve organisatie in het voor- en natraject);
- . een consistente dossierorganisatie.

Bijzondere opdrachten

Naar aanleiding van het ooit benoemen van een AC-vennoot als arbiter bij geschillen tussen twee contractanten in de automatiseringssfeer en het feit dat het verschijnsel getuige-deskundige uw verslaggever uit zijn privé-sfeer niet vreemd is, werd ook naar dit soort zaken summier onderzoek ingesteld.

Gebleken is, dat in de USA daadwerkelijk enige malen een EDP-auditor (vennoot van de Amerikaanse KMG-partner) als getuige-deskundige bij rechtszaken is ingeschakeld.

In Canada kwam (toevalligerwijs tijdens mijn bezoek) het verzoek van een softwarehouse om hun boekhoudpakket te bestuderen teneinde te anticiperen op vragen die een mogelijk toekomstige klant van dit softwarehouse hierover zou stellen.

Auditing IBM's OS-operating systems

In deze paragraaf volgt een weergave van enkele van de bijgewoonde sessies tijdens de twee grote AC-vakconferenties.

Michael I. Sobol (MIS Associates) leidde een 2 dagen durende bijeenkomst onder de titel "Facilities for auditing IBM's OS/VS operating systems". De behandeling ging in eerste aanleg in op "introductie OS" en de job control language en vervolgens met name op het gebruik door de EDP-auditor van de beschikbare utilities.

In het bijzonder werd het nut van de utility IEHLIST toegelicht; de option LISTCTLG voor het uitprinten van de system catalog (wat staat op welk pack), de option LISTVTOC (wat staat waar op het pack; interessant zijn bijvoorbeeld de partitioned data sets -pds- met daarin opgenomen programma's) en de option LISTPDS voor het uitprinten van de pds-directory (namen, lengtes en adressen van programma's). Gewijzigde lengtes van pds-members kunnen duiden op programma-wijzigingen (dit behoeft niet noodzakelijkerwijs het geval te zijn geweest: de compress-functie kan zijn toegepast). Vervolgens zouden met behulp van de utilities IEBTPCH (print-punch) of IEBGENER pds-members kunnen worden afgedrukt.

Maakte Sobol in een positief te noemen zin gebruik van de utilities,

Jeffrey Ferrey deed dit in zijn inleiding op een negatief te noemen, doch vermoedelijk effectieve wijze. In zijn inleiding over OS-audit gaf hij een soort "kookboekrecept".

Hij startte zijn audit met interviews, bestuderen van documentatie en procedures ten einde vast te stellen welk OS (versie) in gebruik was, welke afwijkingen van het standaard OS bestonden, wie verantwoordelijkheden droegen en hoe deze uitgevoerd werden met name ten aanzien van access en het monitoren hiervan. Onder meer richtte hij zich op access van catalogs, system data sets en libraries.

Vervolgens gaat hij een en ander testen en dan loopt het recept als volgt verder:

- a. Execute the IBM utility program IEHLIST/LISTCTLG against the SYSCTLG dataset. If this job executes successfully, access to the system catalog is not adequately controlled.
- b. Execute the IBM utility program IDCAMS/LISTCAT against the Master Catalog and User Catalogs. If any of these jobs executes successfully, access to the Master Catalog/User Catalogs is not adequately controlled.
- c. Execute the IBM utility program IEHLIST/LISTPDS against SYS1.NUCLEUS, SYS1.UADS and SYS1.PROCLIB. If any of these jobs executes successfully, access to the operating system datasets is not adequately controlled.

Zo gaat het nog een tijdje door. Gebruik makend van IEH- en IEB-utilities alsmede van SUPERZAP worden allerlei libraries en datasets benaderd, die in feite terdege beschermd zouden moeten zijn.

Zeer gemakkelijk is vervolgens:

- k. If any of the above jobs using IEHLIST executes successfully, access to the IEH-utilities is not adequately controlled. (IEH-utilities zijn gevaarlijke en derhalve af te schermen utilities.)

Ferrey claimt met deze aanpak meermalen succes te hebben gehad en uit zelfs de verwachting dat in 90% van de gevallen waarin deze methode wordt toegepast van succes sprake zal zijn.

William Konigsford werkt aan het opstellen van Standards for Operating Systems Security volgens een systematiek vergelijkbaar met de Computer Control Guidelines van het CICA. Hij baseert zich hierbij op de Control Subjects uit de TRW/IBM-studie Data Security and Data Processing.

De volgende Control Subjects worden onderkend:

- . Access Control
- . Identification
- . Isolation
- . Surveillance
- . Integrity

Vervolgens worden per Control Subject gedefinieerd:

- . Control Objectives
- . Control Standards
- . Control Requirements, inclusief een poging "levels of optionality" toe te kennen (must, shall, may, might).

Het geheel is op dit moment nog niet uitgekristalliseerd. Konigsford noemt het zelf een "first step" naar een systematische benadering.

Slot

In het besef dat vele zaken nader kunnen worden uitgewerkt dan wel hier in het geheel niet aan de orde komen, is volstaan met een globale weergave van een greep uit de veelheid aan AC-onderwerpen waarmee ik tijdens de reis in aanraking ben gekomen. Over het geheel genomen heb ik de reis als nuttig ervaren en het proeven van een "andere wereld" als boeiend.

Amsterdam, 13 augustus 1981.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

COMPUTERTOEPASSING: VOORRAADWAARDERING

door drs. J.E. Huizenga

Inleiding

In de serie "Computertoepassingen" deze keer een beschrijving van de toepassing van het CARS-3-programmapakket. CARS is een COBOL-programma dat met behulp van parameterkaarten en een bestand met "losse" COBOL-zinnen een nieuw COBOL-programma genereert. Dit gegenereerde programma (AUDIT-source genaamd) is na vertaling geschikt om het te onderzoeken bestand (of maximaal zeven bestanden) te verwerken (zie bijlage 1). De uitvoer kan zijn in de vorm van:

- bestandstotalen (automatisch);
- totalen van maximaal 99 selecties;
- maximaal negen onafhankelijke overzichten op grond van selectiecriteria;
- een overzicht van posten geselecteerd op grond van steekproefcriteria;
- een uitvoerbestand.

In de hier beschreven toepassing wordt van de laatste mogelijkheid geen gebruik gemaakt.

De cliënt

De hier beschreven computertoepassing maakt deel uit van de jaarrekeningcontrole bij een cliënt die een groothandelsfunctie vervult. Een belangrijk deel van de computercapaciteit wordt besteed aan sturing en registratie van de goederenstroom. De voorraadregistratie geschiedt op basis van per maand berekende verrekenprijzen. De verrekenprijs komt tot stand door de bepaling van de gemiddelde inkoopprijs van de liggende voorraad volgens de FIFO-methode, vermeerderd met opslagen voor magazijnkosten en een per artikel wisselend opslagpercentage voor vrachten en rechten.

Het controleprobleem

Het zal voor Compact-lezers geen verrassing zijn dat de accountant belast met de jaarrekeningcontrole belangstelling heeft voor de consistentie van de toepassing van de bovengeschetste waarderingsregel. De interne controle op de factoren welke invloed hebben op de waardering van de voorraden per ultimo (controle op programmawijzigingen, correctieprocedures en dergelijke) zijn in dit geval zodanig dat een zelfstandig oordeel over de uitkomsten van de waarderingsprocedure gewenst is. Zoals een groothandel betaamt, maakt het aantal artikelen een integrale controle onaantrekkelijk. Het lag voor de hand voor de oplossing van het probleem gebruik te maken van de mogelijkheden van een audit-package.

De computertoepassing

Onderzoek van het geautomatiseerde voorraadbeheersings- en registratiesysteem leerde onder andere dat het artikelstambestand een schat aan informatie bevat (bij een recordlengte van 900 bytes niet zo verwonderlijk).

Voor het waarderingsprobleem zijn bijvoorbeeld van belang:

- hoeveelheid, waarderingsprijs en waarde primo boekjaar;
- hoeveelheid, waarderingsprijs en waarde ultimo boekjaar;
- cumulatieve inkopen, hoeveelheid en waarde tegen waarderingsprijs in het boekjaar;
- cumulatieve verkopen, hoeveelheid en waarde tegen waarderingsprijs in het boekjaar;
- laatste inkoopdatum en inkoopprijs;
- percentage vrachten en rechten;
- leveranciersnummer (elk artikel heeft slechts één leverancier).

Het inkoopbeleid van de cliënt leidt ertoe dat van de liggende voorraad van een artikel normaliter een hoog percentage administratief (FIFO!) tot de meest recente inkoop wordt gerekend.

Het ligt dus voor de hand te veronderstellen dat in normale gevallen de waarde van de eindvoorraad van een artikel berekend volgens het systeem van de cliënt niet veel zal afwijken van een waarde welke als volgt gesimuleerd is:

Waarde = hoeveelheid x (laatste inkoopprijs + opslagen magazijnkosten, vrachten en rechten).

De kern van de computertoepassing is dan ook het signaleren van die artikelen die meer dan een kritisch verschil vertonen tussen de gepresenteerde en de werkelijke waarde, dan wel waarvoor de waarderingsprijs meer dan een bepaald percentage afwijkt van de voor magazijnkosten, vrachten en rechten gecorrigeerde laatste inkoopprijs. Zie bijlage 2 voor de af te drukken gegevens per geselecteerd artikel.

Uiteraard is deze controletechniek slechts aanvaardbaar indien redelijke zekerheid bestaat dat de gegevens welke in de simulatie worden gebruikt, met name de laatste inkoopprijs, betrouwbaar zijn. Om die zekerheid te krijgen, wordt tevens een steekproef getrokken uit het artikelbestand per ultimo boekjaar. Van de geselecteerde artikelen wordt via de crediteurenadministratie onderzocht of de laatst geregistreeerde inkoop inderdaad de laatste van dat boekjaar is en of de gegevens juist in het artikelstambestand zijn opgenomen.

Verder dient uiteraard door totaaltellingen te worden vastgesteld dat het onderzochte bestand gelijk is aan het bestand dat werd gebruikt om de post voorraden in de jaarrekening samen te stellen. Voor een schematische samenvatting, zie bijlage 3.

Besluit

De hier beschreven toepassing is reeds een aantal jaren in gebruik en is een belangrijk hulpmiddel bij de jaarrekeningcontrole. Inmiddels is de toepassing uitgebreid met:

- een lijst waarop de oude voorraden met een boekwaarde boven een bepaald bedrag worden afgedrukt;
- een stratificatie van de voorraden per afdeling naar ouderdom.

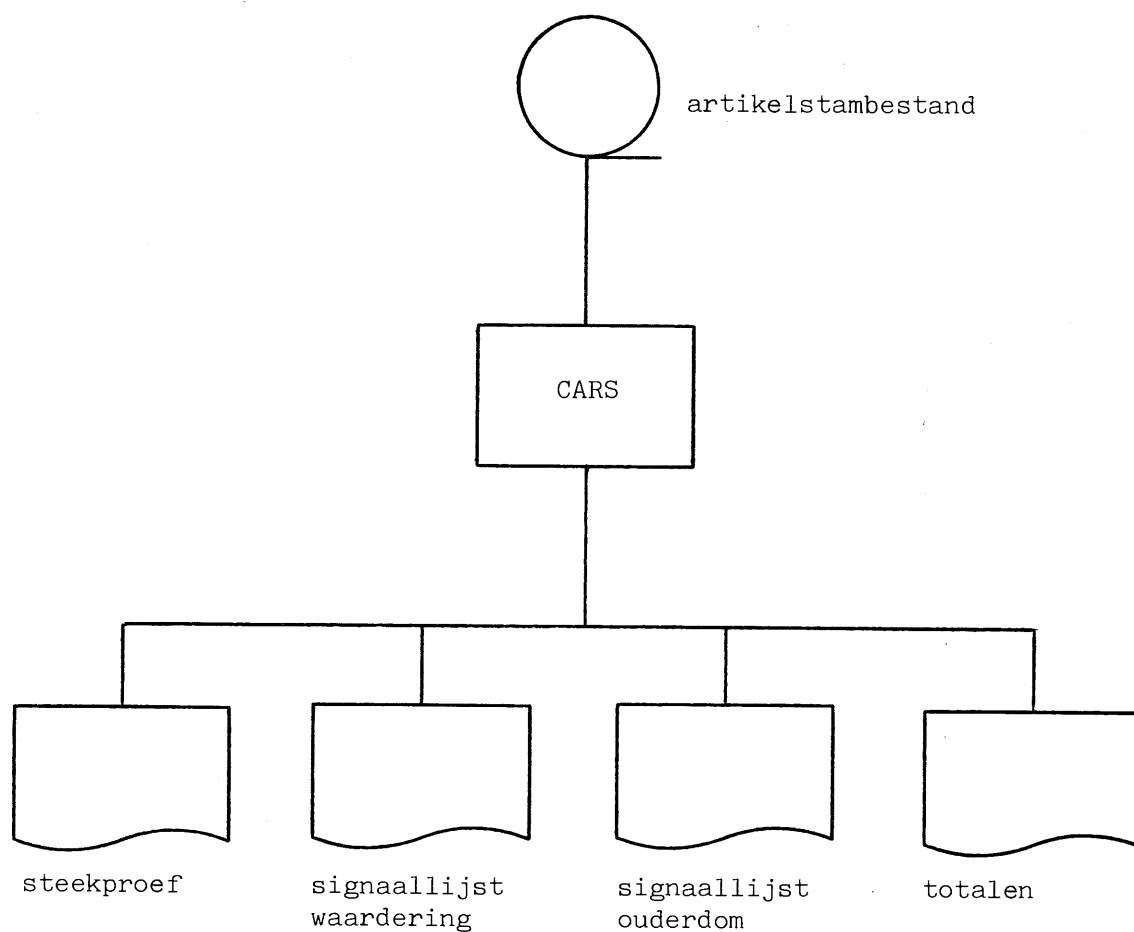


COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

Beschrijving afdruk

Per geselecteerd artikel wordt afgedrukt:

- afdelingsnummer,
- artikelnummer,
- fabrikantenummer,
- artikelomschrijving,
- laatste inkoopdatum,
- laatste verkoopdatum,
- waarde beginvoorraad,
- waarde cumulatieve verkopen,
- waarde cumulatieve inkopen,
- hoeveelheid eindvoorraad,
- waarde eindvoorraad,
- waarderingsprijs beginvoorraad,
- gemiddelde inkoopprijs,
- waarderingsprijs eindvoorraad,
- laatste inkoopprijs,
- gesimuleerde waarderingsprijs eindvoorraad,
- waardeverschil tussen eindvoorraad tegen de waarderingsprijs en de gesimuleerde waarderingsprijs.



DE BETEKENIS VAN DE PROGRAMMEURSSECTIE VOOR AUTOMATISERING
EN CONTROLE

=====

door H. Weerd

Inleiding

De controlerend accountant wordt vrijwel continu geconfronteerd met veranderingen in de administratieve organisatie van zijn cliënten, niet in het minst op het gebied van de geautomatiseerde gegevensverwerking. Indien door een geautomatiseerd proces van gegevensverwerking informatie wordt verstrekt welke invloed heeft op de jaarrekening, zal de controlerend accountant zich een oordeel wensen te vormen over de wijze waarop deze informatie tot stand komt.

De veranderingen in de administratieve organisatie kunnen de accountant aanleiding geven om wijzigingen aan te brengen in zijn controleplan en om bij de uitvoering van de controle eventueel gebruik te maken van de computer. Door van deze technische mogelijkheid gebruik te maken kan de accountant zijn controle efficiënter uitvoeren en zal in een aantal gevallen de kwaliteit van zijn controle kunnen worden verbeterd.

Werkzaamheden van de programmeurssectie

De werkzaamheden van de programmeurssectie vallen globaal uiteen in de volgende delen:

1. Het aan de hand van door een controleploeg opgestelde wensen samenstellen en onderhouden van computerprogramma's als hulpmiddel voor de controle van de jaarrekening;
2. het periodiek uitvoeren van deze programma's op de computer;
3. het verzorgen van cursussen in het kader van de AC-opleiding;
4. het uitvoeren van research-werkzaamheden.

Aan de hand van deze indeling wordt in dit artikel uiteengezet op welke wijze de accountant van deze werkzaamheden gebruik maakt.

Het samenstellen en onderhouden van computerprogramma's als hulpmiddel voor de controle van de jaarrekening

De programmeurssectie is primair belast met het ontwikkelen en onderhouden van toepassingen met behulp van hogere programmeertalen (Cobol en RPG II), ingeval deze niet of niet efficiënt met behulp van de binnen ons internationaal samenwerkingsverband (KMG) gehanteerde standaardprogrammatuur kunnen worden ontwikkeld. De standaardprogrammatuur bestaat uit:

- CULLPRIT/EDP-AUDITOR
- CARS

Om de overdraagbaarheid van controleprogrammatuur binnen de programmeurssectie te vereenvoudigen zijn standaards ten aanzien van documentatie en codering vastgesteld.

Om periodiek te kunnen beschikken over gegevensverzamelingen van de cliënt dienen in de ontwikkelingsfase afspraken te worden gemaakt, welke vervolgens schriftelijk worden bevestigd. In geval de verwerking zal plaatsvinden op de computer van de cliënt worden hiertoe eveneens afspraken gemaakt.

Na het testen en in gebruik nemen van de toepassing worden van de documentatie twee sets vervaardigd, waarvan een set bij de dossiers van de controleploeg wordt bewaard ten behoeve van reconstructie mogelijkheden. Deze documentatie maakt tevens deel uit van het controleprogramma.

Het periodiek uitvoeren van computerprogramma's op de computer

Het beleid van de maatschap, met betrekking tot het uitvoeren van computercontroleprogramma's, is er -met het oog op de te bereiken hoge mate van onafhankelijkheid in de uitvoering van het controleprogramma- op gericht de verwerking zoveel mogelijk op de eigen computerinstallatie te laten uitvoeren.

Indien de verwerking niet op de eigen (KKC) computer kan geschieden, dient de ongestoorde verwerking van de computercontroleprogramma's zoveel mogelijk te worden gewaarborgd door de aanwezigheid van AC-programmeurs in het rekencentrum van de client.

De werkzaamheden van de AC-programmeur bestaan in dit geval uit:

- plannen van de uit te voeren werkzaamheden;
- registreren van de informatiedragers met de te verwerken gegevensverzamelingen;
- gereed maken van de besturingskaarten;
- het inlezen of toezien op het inlezen van besturingskaarten en programma's in source of object vorm;
- het volgen van de activiteiten op het systeemconsole gedurende de uitvoering van de controleprogramma's en het afdrukken van de overzichten;
- het toezien op correcte toewijzing van de vooraf geregistreeerde informatiedragers (magneetbanden en disks) aan de programma's;
- het in ontvangst nemen van magneetbanden, waarop standen en/of tussenresultaten zijn opgeslagen;
- het in eigen beheer bewaren van vorengenoemde bestanden;
- het in ontvangst nemen en verzenden van de vervaardigde overzichten;
- het archiveren van de uitvoerings/besturings overzichten;
- het onderhouden van contacten met de controleploeg.

Deze activiteiten kunnen worden aangevuld met programma-technische hulpmiddelen met als doel te voorkomen dat de uitvoering van controle programmatuur wordt beïnvloed. In het kader van een afstudeeropdracht van een stagiaire van de HTS-informatica te Eindhoven is een programma ontwikkeld met als functie het berekenen van een

hashtotaal van de uit te voeren instructies van de controleprogramma's, opdat vastgesteld kan worden dat de programmatuur in ongewijzigde vorm in het geheugen wordt geladen, voordat de uitvoering van de desbetreffende programmatuur plaats vindt. Deze methode is nog niet algemeen toepasbaar. (referentie 1)

Verdere research-werkzaamheden op dit gebied vormen naar mijn mening een belangrijke bijdrage aan de beveiliging van de uitvoering van controleprogrammatuur door accountants.

Het verzorgen van cursussen in het kader van de AC-opleiding

In de AC-opleiding wordt een belangrijk deel van de praktijkonderdelen met betrekking tot programmeren door leden van de programmeurs sectie verzorgd. Van hun specifieke kennis op het gebied van programmeren wordt bij dit onderdeel van de AC-opleiding gebruik gemaakt. Dit om de AC-accountant-in-opleiding kennis en begrip bij te brengen over de wijze waarop gegevens in een toepassingsprogramma door een geautomatiseerd systeem worden verwerkt.

Research werkzaamheden

De controlerend accountant die de opzet en werking van de automatiseringsorganisatie wenst te toetsen, zal in toenemende mate worden geconfronteerd met het gebruik van pakketten van programmatuur voor de uitvoering en beheersing van het productieproces in het rekencentrum.

De NivRA-studie en de AC-opleiding geven de desbetreffende accountant in deze situaties niet in alle gevallen voldoende technische kennis om uit de massale detailinformatie de specifieke controle-aspecten te onderscheiden. Ook in deze situatie kan de ervaren programmeur de controlerend accountant van dienst zijn door uit de detailinformatie, die met betrekking tot het desbetreffende pakket beschikbaar is, de specifieke controle-aspecten samen te vatten. Vervolgens kan de controlerende accountant aan de hand van de opgesomde controle-aspecten en de gegevens over de specifieke toepassing bij zijn cliënt het gebruik van het desbetreffende pakket evalueren.

Huidige samenstelling van de programmeurssectie

In zijn huidige samenstelling bestaat de programmeurssectie uit:

- junior-programmeurs (11)
- programmeurs (2)
- AC-kundigen (4)

Met de dagelijkse leiding is een EDP-auditor vrijwel full-time belast.

Bij de werving van junior-programmeurs is het beleid er op gericht dat de leden van de sectie kunnen doorgroeien naar de functie van EDP-auditor. Afhankelijk van hun verdere theoretische vorming kunnen zij zich specialiseren in de richting van Controle (NIVRA-studie) of automatisering (AMBI-studie).

Conclusie

De programmeurssectie van de AC-kern vervult op verschillende gebieden van de accountantscontrole binnen onze maatschap een rol van betekenis. Aan de gereleveerde activiteiten ontleent de sectie zijn bestaansrecht.

Referentie

1) Zekerheid bij de verwerking van controleprogrammatuur, Amsterdam
15 mei 1981, A.J. Henriquez.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

BOEKEN

=====

Door J. Filippo

AC 291 Effective Methods of
EDP Quality Assurance - William E. Perry

Quality Assurance dient - als intern controlemiddel- D.P.-management de zekerheid te verschaffen, dat systemen worden ontwikkeld als een kwaliteitsproduct. Het boek verschaft een methodologie voor de invoering van de Quality Assurance functie binnen de D.P.-organisatie met het doel (D.P.-)management een onafhankelijke evaluatie van de doeleinden, methoden en uitvoering (zie fig. 1) in alle fasen van de systeemontwikkeling te verschaffen.

Fig. 1 Elements of objectives QA-function

GOALS	METHODS	PERFORMANCE
Meets user needs	The system was implemented using organization and EDP department: Policies Procedures Standards Guidelines	The system design is : Economical Effective Efficient
Consistent with needs of other users		
Consistent with organization objectives		
Meets EDP department objectives		
Consistent with industry and government requirements		

Het boek bevat vier secties, namelijk

1. de Quality Assurance function :
de essentie, de rol en bemanning
2. de planning van de QA-functie :
systeem review, application review
3. de QA-reviews :
initialisering, uitvoering, technieken en rapportage
4. speciale taken en de relatie met internal auditing appendix :
voorbeeld van een QA manual.

Een gedetailleerde bespreking van deze secties zou in het kader van Compact te ver voeren, zodat volstaan zal worden met enige punten.

De bemanning van de QA-functie zoekt de schrijver in mensen met ervaring op het gebied van systeemontwikkeling. Afhankelijk van de omvang van het QA-werk varieert de bemanning van full-time tot task-force. De EDP-manager is in eerste instantie de leider van de QA-groep.

Tot de taken van de QA-groep behoren (in deze volgorde) :

- het ontwikkelen van standards en guidelines voor de EDP-organisatie;
- review van de general controls;
- verschaffen van technische adviezen;
- review van de ontwikkeling van applicaties (in alle fasen).

In het algemeen ziet de schrijver de QA-functie als een verlengstuk/staffunctie van EDP management en zij kan dus op allerlei plaatsen worden ingezet, mits dat niet de onafhankelijkheid aantast.

Ten aanzien van de interne accountants-functie onderkent de schrijver dat deze enerzijds andere, anderzijds overlappende taken taken heeft in vergelijk met de QA-functie.

Bij het functioneren van de QA zal de (EDP) accountant een deel van het werk, dat tijdens de systeemontwikkeling werd gedaan, achterwege kunnen laten en gebruik kunnen maken van het werk van de QA-functie (als interne controle functie). Hierdoor zal de interne accountant ook meer onafhankelijk staan ten aanzien van de systeemontwikkeling.

Voor een nauwe samenwerking tussen IAD en QA wordt gepleit, zelfs met de suggestie van wederzijdse uitwisseling van personeel met het doel van training.

Buiten het aspect van de QA-functie wordt hoofdstuk 12 van het boek, dat over de verhouding IAD-QA handelt, ter lezing aanbevolen voor de duidelijke behandeling van de interne (EDP) audit functie.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

L ITERATUUR

door J.C.P.M. Vermeeren en drs. B.M. de Vries

Geïntegreerde kantoorssystemen
De Accountant juni 1981

- J.G.T.H. Rief

=====

Steeds vaker worden we geconfronteerd met publicaties over het onderwerp kantoorautomatisering, office automation of kantoor van de toekomst. Daarbij wordt gedoeld op een stroming in de automatisering van kantoorwerkzaamheden, die gericht is op communicatie en informatie behandeling.

Het begrip geïntegreerd kantoorstelsel wordt door Rief gedefiniëerd als:

"het in samenhang gebruiken van elektronische apparatuur voor de hantering van informatie, zowel gestructureerd, niet-gestructureerd als in de vorm van afbeeldingen, in een organisatie".

De kantoorautomatisering wordt door de volgende factoren gestimuleerd:

- toename van het aantal arbeidskrachten in de indirecte sector;
- afname apparatuurkosten;
- toename productiviteit door automatisering;
- toename van de informatiebehoefte.

Voorts hebben de technologische ontwikkelingen op het gebied van micro-elektronica (chip), van de communicatietechniek (glasvezel, communicatiesatelliet) en van de vastlegging van informatie (bubble memories, video-disk) de ontwikkeling in de richting van geïntegreerde kantoorssystemen gunstig beïnvloed.

Het veranderingsproces betreffende kantoorautomatisering wordt door de auteur met behulp van schema's toegelicht. Daarbij wordt uitgegaan van een ontwikkeling die begint met tekstverwerking, vervolgens wordt uitgebreid met toepassing computer apparatuur voor administratieve doeleinden en de daarmee verbonden terminal-netwerken.

De organisatorische aspecten van een zodanige ontwikkeling worden aan de hand van modellen vrij cryptisch behandeld. Het is mogelijk, dat de kantoorautomatisering tot ver doorgevoerde decentralisatie zal leiden, waarbij de draagbare terminal de mogelijkheid biedt tot het verrichten van werkzaamheden thuis.

Enerzijds opent dit mogelijkheden voor deeltijdbanen en behoeft het kantoor minder frequent te worden bezocht (dus minder reistijd, benzineverbruik, files, milieuverontreiniging). Anderzijds kan het thuiswerken met de terminal isolerend gaan werken. De schrijver denkt derhalve aan een tussenoplossing, de Buurtkantoren.

Countdown Services: FIRE and its AFTERMATH
in a Computer bureau

- N.M. Duffy

Information & Management september 1980. (Bron: EDPACS april 1981).

=====

Fire!

In the early evening of April 26, 1977, the smoke and fire detection devices in the computer room at Countdown Services were triggered. The curtains and ceilings in some adjoining offices were burning. The operations supervisor grabbed a fire extinguisher, but smoke kept him from entering the computer room. The fire department telephoned at this point to confirm the alarm. The supervisor shut down the power and began to direct the removal of some tape and disk files. He also started calling senior members of management. In less than 15 minutes the operations manager was on the scene. He had the most important tapes and disks loaded into his car. Fortunately, the online system had been shut down and the data base had been dumped prior to the fire. As a result, although some of it was slightly damp, backup was available and had been saved. Water was pouring through the airconditioning ducts onto the CPU and tape and disk units. Thick, acid smoke billowed through the installation. Crawling on their hands and knees to avoid the smoke, the DP staff removed all 2000 tapes from the library. There were problems because the firemen were not knowledgeable about how to handle computer fires. By one o'clock in the morning, the fire was finally out. Removal of material continued until 1:30 a.m. Tapes were being stacked and sorted in the boardroom. Calls began going out to other Burroughs computer users around the country.

Background

The victim of this fire, Countdown Services, is a subsidiary of South Africa's largest retail automobile dealer. At the time of the fire, the service bureau was using a Burroughs B-4700 mainframe to operate an online terminal network and provide batch processing. Peripheral devices included four dual disk drives, eight tape drives two printers, a card reader, and a front-end communications processor.

Aftermath

Prior to the fire, the company had implemented a number of security procedures:

- Installation of fire detectors, extinguishers and hoses;
- Offsite storage of software in a fireproof safe;
- Return of source documents to the originating companies immediately following data entry;
- One-month retention of input data;
- Onsite storage of first generation backup files;
- Offsite storage of the second generation file backup;
- Offsite storage of enough preprinted output forms to satisfy vital requirements until replacement stock could be obtained;
- Preparation and practice of procedures to be followed in the event of a fire.

After the fire the company found it had a lot to do. Some of the actions were:

- Contacted Burroughs to supply replacement hardware;
- Located alternate processing facilities. (Nothing was available locally. Installations in two other cities could be used, but operators, tapes, and disks had to be flown around the country)
- Informed users, accountants, bankers, suppliers and other interested parties about the fire and its implications.

Major Problems

Two problems deserve special mention:

1. Staff Morale

Members of the EDP operations staff were worried about their jobs. It required direct assurances from management to convince the staff that they would continue to be employed. Further, long periods of time away from home and irregular schedules for sleeping and eating had a negative impact on control.

2. Processing at Remote Locations

Processing at other installations in remote locations is fraught with frustrations. Because of unavoidable scheduling conflicts, much of the processing had to take place at nights and on weekends. Despite the full cooperation of the staff at both alternate sites, it was very difficult to get enough processing time. Interruptions were frequent. As a result, operating standards deteriorated. For example, the frequency of file dumps was reduced.

Because the alternate sites were in two different cities, tapes sometimes ended up in the wrong place. Since the experienced staff could not be in three places at one time, it was hard to balance processing runs and maintain quality control.

Return to Normal

Almost one month after the fire, a new installation was operational. Some of the tapes and disk files were not compatible, but Countdown Services had a new home.

After a while some of the tapes that survived the fire began to fail and had to be replaced. The new EDP site was subject to power surges and cuts, so operations were often interrupted. Systems and operations were at different locations.

Although it may sound like a minor point, a lack of adequate parking at the new site caused staff morale to drop even lower.

Lessons to be learned

Based on its experience, Countdown Services has some advice for other EDP installations:

- Review insurance coverage. Pay particular attention to the added costs of working on different pieces of hardware;
- Take a close look at provisions for backup. Running in backup is far more difficult than anyone can imagine;
- Consider detailed security provisions. What appears adequate may be found wanting when put to test.

Summary

This article provides a very interesting and informative case study of an EDP fire and the problems of recovery. The original text contains a number of very useful figures, exhibits, and tables, as well as descriptions of other unanticipated problems.

Organisaties gaan steeds meer over op het gebruik van gegevensbanken. Een van de kenmerken van gegevensbanken is dat er een ont-koppeling heeft plaatsgevonden tussen de opslag en het gebruik van gegevens. In de praktijk betekent dit, dat iemand de gegevens vastlegt en aan de computer aanbiedt voor opslag en dat een ander of anderen de gegevens gebruikt/gebruiken.

In deze situatie wordt de noodzaak gevoeld nog scherper dan voorheen vast te stellen wie eigenaar en wie gebruiker is. Deze noodzaak vloeit voort uit de wens van de gebruikers, dat gegevens volledig, juist en up to date zijn. Ook dienen de gegevens tegen onbevoegd gebruik te worden beschermd en dient het vóórkomen van synoniemen en homoniemen te worden bestreden. Op de vraag wie verantwoordelijk is voor gegevens zal derhalve een duidelijk antwoord gegeven moeten worden. Voor de aanduiding van de verantwoordelijkheid ten aanzien van de gegevens zal eerst het begrip "de gegevens" nader moeten worden ontleend. Onderscheid dient te worden gemaakt tussen metagegevens (de beschrijving van gegevens = data type) en actuele gegevens (data occurrence).

Volgens Völlmar is de verantwoordelijkheid op te splitsen in een aantal verantwoordelijkheden, zoals in onderstaande tabel is aangegeven:

De verantwoordelijkheid voor gegevens in een organisatie valt bij analyse uiteen in een aantal verantwoordelijkheden.

	METAGEGEGEVENS			AKTUELE GEGEVENS		
	inhoudelijk	logische gegevens-structuur	verzameling (data dictionary)	inhoudelijk	technische structuur	verzameling (gegevensbank)
eigenaar	definieert legt vast bepaalt wie mag gebruiken	beschrijft de samenhang in de gegevens	is gebrui- ker	zorgt voor kor- rektheid, volle- digheid en tij- dige invoering		is gebruiker
stysteemont- wikkelaars	assisteren bij inventariseren en beschrijven	bepalen de ge- wenste logische structuur	zijn gebau- wers	ontwikkelen ge- programmeerde kontrolemecha- nismen	doen voorstel- len	
gegevens- bankbeheer- der	reikt systema- tiek aan	biedt ondersteu- ning	zorgt voor beschik- baarheid en af- scherming	assisteert bij het instellen van controle- procedures	beslist over de technische structuur bewaakt de efficiency	zorgt voor be- schikbaarheid, afscherming en efficiency van het gebruik

De beschikbaarheid en afscherming van gegevens (zowel meta- als actuele gegevens) is volgens de auteur de verantwoordelijkheid van de gegevensbankbeheerder, omdat hij "invloed heeft op het gebied van de benodigde technische hulpmiddelen en methoden (data base management system, vaststellen van de technische inrichting van de Banken, reorganisatie en rekonstruktie maatregelen e.d.)". Echter "het criterium waarop de aktualiteit vervalt dient door de eigenaar te worden geformuleerd" en de eigenaar zal moeten aangeven, welke gegevens verwijderd mogen worden. De auteur slaagt er niet in om op overtuigende wijze de verantwoordelijkheid van de gegevensbankbeheerder voor beschikbaarheid en afscherming te argumenteren.

Voor de fundamentele beslissing van het al dan niet beschikbaar stellen van gegevens dient de gegevensbankbeheerder zelfs terug te vallen op de eigenaar.

In het artikel wordt derhalve geen duidelijkheid gegeven over de verantwoordelijkheid voor het beheer van gegevens. Duidelijk onderscheid tussen de beheer- en bewaarfunctie zal aangegeven moeten worden. De beheersdaden, zoals creëren, wijzigen en verwijderen van gegevens dienen tot de verantwoordelijkheid van de eigenaar te behoren. Daarnaast zal de verantwoordelijkheid van de gegevensbankbeheerder aangegeven moeten worden. Hopelijk zal in een door Völlmar aangekondigd artikel de verantwoordelijkheid van gegevensbankbeheerders duidelijker worden.

R. Bron zal in een volgend nummer ingaan op de artikelserie van Völlmar.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

door J.F.C. van Epen en drs. H.C. Kocks

Automatisering

Sinds de aankondiging van het IBM-programmaproduct SQL/DS (zie Compact nr. 23, Lente 1981) wordt de wedren met relationele data base systemen opgevoerd. Het Deense Tom Pedersen International heeft de rechten voor Europa van het relationele data base management systeem Oracle. Het door het Amerikaanse Relational Software Incorporated ontwikkelde pakket draait op dit moment bij ongeveer zeventig installaties in de Verenigde Staten op PDP- en VAX-systemen van Digital. De eerste implementatie op IBM-apparatuur is bij een grote Amerikaanse bank een feit en Tom Pedersen verwacht medio herfst aanstaande een definitieve versie voor IBM's besturingssystemen OS en DOS.

Aldus de inleiding van het artikel in Computable van 29 mei 1981:

Deens systeemhuis brengt relationele DBMS in Europa

Oracle sinds 1979 op DEC; binnenkort ook voor IBM

Het Deense systeem- en programmatuurhuis bestaat vijf jaar en het eerste contract betrof een automatiseringsopdracht van de regering van Koeweit. Het met 35 medewerkers - vrijwel allemaal academisch geschoold - opererende bedrijf levert volgens Bo Royden, technisch directeur en mededirecteur samen met Tom Pedersen, projecten tegen vaste prijs en tijd - "turn key" - waaronder het Deense persoonsregistratiesysteem. "De marketing van Oracle voor de Europese markt gebeurt pas sinds een maand", volgens Royden. "Wij zijn druk bezig onder meer een Nederlandse vertegenwoordiging voor Oracle te vinden. Inclusief technische ondersteuning."

Oracle bevat drie hoofdcomponenten. De Sequel 2 opvraagtaal, een geïntegreerde gegevenscatalogus (data dictionary) en de nucleus van het data base management systeem "Oracle Kernel".

Sequel 2 is een hogere Engelstalige opvraagtaal, oorspronkelijk door IBM ontwikkeld voor het nog niet geannonceerde Systeem R (de veel gebruikte naam voor een door IBM te ontwikkelen relationeel data base systeem, red.). Sequel 2 bevat meerdere opvraagniveaus, elk met een ander complexiteitsniveau. Alle gebruikers van Oracle - de eindgebruiker, de specialist of "data administrator" - maken op hun eigen niveau gebruik van dezelfde opvraagtaal. Sequel 2 ondersteunt terminalbesturing, een subtaal voor procedurele talen zoals FORTRAN, COBOL en PL/1, een rapportgenerator en faciliteiten voor gegevensdefinitie en -controle in een gegevenscatalogus. In de geïntegreerde "data dictionary" en "directory" definieert Oracle dynamisch alle tabellen. Gegevensstructuur en -organisatie zijn transparant voor de gebruiker. Toegang tot de opgeslagen gegevens krijgt de gebruiker door het definiëren van zijn kijk - view - op de tabellen en het leggen van relaties tussen kolommen uit tabellen.

De gegevensstructuur van de data base is volledig onafhankelijk van de applicatieprogrammatuur. De gegevenscatalogus biedt de data-definitie dynamisch aan op het moment dat het programma de gegevens benadert. Wijzigingen in fysiek geheugen of benaderingsmethoden zijn transparant voor de gebruikersprogrammatuur. Hetzelfde geldt bij logische bestandswijzigingen en uitbreiding van de totale data base. De derde hoofdcomponent Kernel omvat efficiency en beschermfaciliteiten. Kernel biedt onder meer compressietechnieken om schijfruimte te besparen, naast een relationele benadering ook gecomprimeerde indexen (inverted list) en ondersteuning van multitaskverwerking voor zowel stapelverwerking (batch) als directe - online - verwerking. Alle data wordt beschermd. De Oracle nucleus bevat integriteits- en consistentieroutines en faciliteiten voor autorisatie, herstart en herstel op basis van een controlepunt - checkpoint - bestand. De Oracle-processor kan alleenstaand worden gebruikt of als achtergrond of "bank-end" machine in een configuratie waarin het centrale systeem moet worden ontlast. De processor waarop Oracle draait kan tegelijk actief zijn op meerdere data bases.

Oracle inclusief de Interactive Application Facility (IAF) voor programmageneratie in onder andere COBOL en FORTRAN vereist 120 kilobytes intern geheugen. Het DBMS staat in de taal C en is volledig "re-entrant" - gelijktijdig te gebruiken door meerdere taken - geschreven.

Dat men, ook al zien de produkten er nog zo mooi uit, voorzichtig en weloverwogen te werk moet gaan bij de aankoop van standaardprogrammatuur is uiteengezet in een artikel in De Automatisering Gids van 10 juni 1981 van Stephen V. Wright, consultant in de Verenigde Staten, die zich heeft gespecialiseerd in het werken met de data base management systemen, onder de titel:

Vooraf bij Data Base Management Systemen:

Oppassen bij aanschaf nieuwe programmatuur

Hoewel de jaren tachtig wellicht het tijdperk zullen worden van de aangekochte programmatuur, zullen zij ook het tijdperk worden van het voorzichtig aankopen van programmatuur, vooral in een verwerkingsomgeving van een Data Base Management Systeem.

Consultants, leveranciers en eindgebruikers zijn zich allen bewust van de voordelen, die men kan verkrijgen wanneer men gebruik maakt van aangekochte toepassingsprogrammatuur. Deze voordelen hebben ook betrekking op het geld, dat men bespaart doordat men niet zelf opnieuw het wiel hoeft uit te vinden.

Maar wat is er bekend over de onzekere factoren van gekochte programmatuur, die wordt toegepast in combinatie met een Data Base Management Systeem?

Het doelmatig verwerken is de eerste factor die men dient te overwegen. Vele programmatuurleveranciers beweren tegenwoordig dat hun pakketten beschikbaar zijn voor verschillende Data Base Management Systemen. Een aanzienlijk aantal van deze programmapakketten werd geschreven toen het DBMS-concept nog lang niet overal werd gebruikt. Als een gevolg hiervan werden de meeste van deze toepassingen ontworpen om te werken in combinatie met sequentiële of ISAM gegevensverzamelingen.

Vele leveranciers vinden dat het te duur is om toepassingsprogrammatuur te herschrijven om het systeem te laten werken in combinatie met een Data Base Management pakket. In plaats daarvan vindt er een vervanging plaats van de "gets" en "puts" van de toepassing door de oproepopdrachten van de data base, met gebruikmaking van de datamanipulatietaal van een bepaalde Data Base Management Systeem.

Hierdoor is een betrekkelijk snelle conversie mogelijk, vooral indien de in- en uitvoeropdrachten worden geïsoleerd in een bepaald segment van de toepassing.

In vele gevallen, waarbij het aantal data base records klein is of de dagelijkse, verwerkingsfrequentie betrekkelijk laag ligt, zal dit "snelle en grove" conversieproces betrekkelijk goed werken. Maar de gebruiker kan enkele problemen krijgen met de zeer grote data base systemen, die voorkomen bij veelvuldig gebruikte toepassingen.

Herstel

Het herstel is ook een factor, die men dient te overwegen bij het selecteren van toepassingsprogrammatuur, die moet werken in combinatie met een Data Base Management Systeem. Wanneer een toepassingsstelsel werkt met een sequentieel bestand, vindt het herstel van een systeemonderbreking plaats door middel van het opnieuw laden van een kopie van dat bestand van band naar schijf en het opnieuw starten van dat programma.

Deze herstelmethode is niet beschikbaar bij de meeste Data Base Management Systemen. In plaats daarvan maakt DBMS-programmatuur gebruik van controlepunten, waarbij alle bewerkingen van de data base tijdelijk worden onderbroken. Vervolgens schrijft het systeem alle lopende data base modificaties naar een schijfengeheugen en een magneetband, die de transacties vastlegt. Indien het verwerken wordt onderbroken, dan brengt de data base administrateur de data base terug naar dat controlepunt. Als een gevolg van deze procedure worden de gegevens teruggebracht naar de waarde, die ze vóór het bewerken hadden en naar het tijdstip, toen ze nog geldig waren.

Om dit te laten plaatsvinden, moet het toepassingsprogramma de logica bevatten om vast te stellen dat er een controlepunt wordt genomen. Het probleem is dat een toepassingsprogramma, dat werd omgezet van standaard bestandsverwerking in een Data Base Management Systeem, geen herstellogica kan hebben anders dan voor het opnieuw starten in geval van een systeemonderbreking.

Deze overwegingen betekenen nog niet dat gekochte programmatuur niet dient te worden gebruikt in combinatie met een Data Base Management Systeem of dat alle gekochte programmatuur ondoelmatig wordt omgezet van een stambestand naar een DBMS verwerkingsomgeving. Maar men dient wél voorzichtig te zijn bij het kopen van toepassingsprogrammatuur en zowel te onderzoeken hoe het produkt functioneert als wat het kan doen.

Op de behoefte aan - blijvend - goede software heeft het Nederlandse softwarehouse Data Process ingespeeld, getuige de kop:

Data Process geeft vijf jaar garantie op programmatuur

'Periode technologisch te overzien'

Data Process te Amersfoort heeft besloten om vijf jaar garantie te gaan geven op een deel van de door dit systeemhuis ontwikkelde programmatuur. Deze maatregel vloeit voort uit het jarenlange werken van DP aan kwaliteitsverbetering en de daarmee nauw verbonden standaardisatie. General manager ir. W.G.F. Schellings vertelt: "In feite is het niet helemaal nieuw, tenslotte gaven we altijd al drie maanden garantie. Die duur is opgeschroefd tot vijf jaar. Dat is een periode die technologisch nog wel te overzien is; over vijf jaar zullen er heus nog wel COBOL-programmeurs zijn. Na vijf jaar moet je over het algemeen toch in je software duiken."

Standaardisatie van diverse facetten binnen de software-ontwikkeling leidde tot verhoging van de kwaliteit van het produkt en verlaging van de produktiekosten. Ook de projectbegeleiding geschiedt volgens een vaststaand recept. Voor het administratieve gedeelte van de projectbewaking wordt sinds kort een microcomputer ingezet.

Data Process heeft het reeds aanwezige aantal kwaliteitsfunctionarissen uitgebreid. Deze mensen beoordelen aan de hand van checklisten de programmatuur. De garantie van vijf jaar geldt uitsluitend voor software die voorzien is van het "certified DP-software label". Nu zijn er twee soorten programmatuur. De "in huis" ontwikkelde programmatuur krijgt in principe zonder meer het label.

Daarnaast is er bij de klant vervaardigde software. "Daar hebben we de randvoorwaarden niet in de hand. Maandelijks bezoeken we echter de gedetacheerde medewerkers en controleren de kwaliteit van het produkt. Tussentijds bijsturen is op die manier mogelijk. Ook deze programmatuur kan in aanmerking komen voor het certificaat", aldus Schellings.

Computable, 14 augustus 1981

Dat goede programmatuur niet op zichzelf staat moge blijken uit een artikel in hetzelfde blad, één week eerder, van de hand van Theo Mulder:

Bij aanschaf computer vooraf afspraken maken over:

Prestatievermogen, beschikbaarheid en betrouwbaarheid

In een paginagroot artikel behandelt de heer Mulder de in de titel genoemde aandachtspunten en geeft daarbij aan welke aspecten contractueel vastgelegd dienen te worden om niet het gehele automatiseringsproces één grote teleurstelling te laten worden. De belangrijkste facetten uit dit artikel, die ook voor de externe accountant in zijn adviserende functie van belang kunnen zijn, geven wij hieronder weer.

Wanneer bedrijven een interactieve computer aanschaffen zonder in eigen huis te beschikken over kennis van het selectie- en aanschafproces en zonder eigen automatiseringsspecialisten, die een geavanceerd systeem gedurende een reeks van jaren tot een succes kunnen maken, is zo'n bedrijf in hoge mate aangewezen op de te kiezen automatiseringspartner. Deze partner dient het bedrijf niet alleen door de moeilijke startfase heen te helpen, maar ook stand-by te zijn in de jaren die volgen na de implementatie. Op zich hoeft het bedrijf ook geen automatiseringskennis te hebben, maar hoe kan veilig worden gesteld dat een automatiseringsprodukt wordt gekocht, dat getoetst kan worden op prestatievermogen (performance), beschikbaarheid en betrouwbaarheid. Welke afspraken worden daarover vooraf gemaakt en hoe worden deze contractueel vastgelegd?

Het begint langzamerhand ook buiten de automatiseringswereld door te dringen, dat het levensgevaarlijk is om een "losse" computer te kopen bij een computerfabrikant en de programmatuur te bestellen bij een derde. Door deze gescheiden bestelling is het vrijwel onmogelijk één van de leveranciers aansprakelijk te stellen voor het uiteindelijke totale produkt, dat ontstaat. Wie daarbij programmatuur laat maken zonder eerst voldoende te hebben onderzocht of voor de eigen problematiek al algemene oplossingen in de vorm van softwarepakketten bestaan, loopt een concreet risico een veelvoud voor de programmatuur te betalen.

In dit artikel wordt ervan uitgegaan, dat de verantwoordelijkheid en de uitvoering van de automatisering bij één instantie komt te liggen. Over het algemeen zijn dat niet de (conventionele) computerfabrikanten, maar wordt deze taak meer en meer vervuld door gespecialiseerde systeemhuizen, waarin kennis en ervaring over bepaalde apparatuur, systeemsoftware en automatiseringstoepassingen zijn geïntegreerd. Wanneer een bedrijf onderhandelt met zijn eventuele toekomstige automatiseringspartner zijn er twee situaties mogelijk; er is een externe deskundige (onafhankelijk adviseur) ingehuurd of de organisatie doet rechtstreeks zaken met leveranciers. In beide gevallen ligt de nadruk op de specificatie van het produkt. Wat is de computerconfiguratie, welke functies dan wel programma's behoren tot de te leveren programma-tuur?

Bedrijven die zonder externe deskundige opereren, vragen zelden naar prestatie, betrouwbaarheid en beschikbaarheid. Aan deze begrippen wordt geen inhoud gegeven ofwel men neemt genoeg met uitspraken van commerciële vertegenwoordigers van de automatiseringsleverancier; uitspraken, die achteraf niet meer in de contracten zijn terug te vinden.

Prestatie en responsetijden

Laten we één aspect eens verder uitdiepen. Redelijk is te vragen naar een goede prestatie of performance van het informatiesysteem. Deze prestatie wordt vaak uitgedrukt in een aantal seconden responsetijd bij beeldschermen, maar ook de doorlooptijd van dagelijkse, wekelijkse, enz. batchprogramma's. Iedereen vindt twee seconden responsetijd bij een beeldschermtransactie aanvaardbaar, dus wordt als eis gesteld deze responsetijd te leveren. Vaak wordt uit het oog verloren dat responsetijd een resultante is van apparatuur en programmatuur.

Bij een toenemend aantal tegelijk werkende terminals loopt de responsetijd terug, meestal discontinu. Het vragen naar of beloven van responsetijden zonder dit af te grenzen in aantallen terminals is zinloos. Verder is de responsetijd van een transactie afhankelijk van het aantal bestandsbewerkingen, welke tijdens het invoeren van de transactie dient plaats te vinden.

Zo is dus ook het eisen dan wel beloven van lage responsetijden, terwijl de systematiek van verwerking buiten beschouwing blijft, een inhoudsloze affaire. Het is een zinvolle zaak onderscheid te maken in responsetijden voor verschillende transacties, zoals itemresponse (het afwickelen van één rubriek), schermresponse (binnen één transactie wisselen c.q. vervolgen van schermen), transactieresponse (de tijd tussen het beëindigen en beginnen van twee transacties) en een functieresponse (starten/beëindigen van programmafuncties).

In de meeste automatiseringsovereenkomsten wordt echter niet gerept over responsetijden, waardoor een opdrachtgever zijn aanspraken verliest. Wanneer er wel over wordt gesproken, beperkt het zich vaak tot het vermelden van één responsetijd, hetgeen blijkens het voorgaande bijna op hetzelfde neerkomt. Een goede methode is een afspraak in de vorm van: minimaal x transacties met y beeldschermen in z tijdseenheden bij een ononderbroken beeldschermvoer.

Beschikbaarheidsgraad kwantificeren

Elke computer - van welke fabrikant dan ook - is aan storing onderhevig. Een goed preventief onderhoud, dat op zich tijd in beslag neemt en derhalve de beschikbaarheid vermindert, heeft gunstige invloed op het aantal storingen.

Goede fabrikanten beschikken over MTBF's, storingstabellen met gemiddelde tijden tussen twee storingen. Deze gemiddelden zijn vastgesteld via vele waarnemingen, maar wie garandeert dat juist uw computer zich aan de statistische voorspelling van dat gemiddelde zal houden. Dat komt op hetzelfde neer als het beweren dat men in een rivier met een gemiddelde diepte van 1,50 meter niet kan verdrinken. Beschikbaarheid tussen de 94 en 100% is normaal, maar dat is geen waarborg voor het uitblijven van een stilstand van uren of dagen.

Betrouwbaarheid onbekend

Laten we betrouwbaarheid van een informatiesysteem definiëren als het geheel van voorzorgsmaatregelen ten einde te bewerkstellingen, dat een informatiesysteem correct en tijdig, die resultaten oplevert, waarvoor het in gebruik genomen is.

Wanneer het negatief wordt benaderd, kan worden gesteld dat een grote foutenkans in de apparatuur, operating systeem, applicatieprogramma's, maar ook in de bediening door beeldschermbedieners en (het nalaten van) handelingen van de systeembeheerder een grote onbetrouwbaarheid met zich meebrengt. Maatregelen met betrekking tot het verhogen van de betrouwbaarheid bewegen zich dus zowel op het terrein van de programmatuurontwikkeling als de organisatie van de informatieverwerking. Ter zake van de programmatuur geldt als norm voor de betrouwbaarheid de mate waarin een programma op abnormale manier kan worden beëindigd. Hoe meer een programma de beeldschermbediende beveiligt tegen fouten en aanwijzingen geeft om in foutsituaties weer tot een normale verwerking te komen, des te betrouwbaarder het programma is. Over deze zaken wordt zelden tevoren overlegd, omdat men zich beperkt tot de functionele specificaties.

De grootste (on)betrouwbaarheid komt echter aan de orde bij het systeembeheer.

Het beveiligen van gegevens is een taak van de systeembeheerder. In die gevallen, waarin het informatiesysteem stagneert (system break down), moet de systeembeheerder een arsenaal van maatregelen hebben om snel weer "in de lucht" te zijn. Het totale systeem is betrouwbaar, wanneer dat lukt in zeer korte tijd en zonder verlies van gegevens. Wanneer in onbetrouwbare systemen vaak moet worden teruggevallen op beveiligde gegevens (back-up's) van een periode terug, maar waarin de systeembeheerder heeft nagelaten beveiligingsacties uit te voeren, is sprake van dramatische dieptepunten van de automatisering.

Tenslotte nog enkele kleine wetenswaardigheden op automatiseringsgebied.

Uit het Financiële Dagblad van 17 juli jl. komt dit berichtje over een elektronische identiteitskaart:

Euro-betaalpas wellicht voorzien van een chip

Verscheidene Europese landen, waaronder Nederland, onderzoeken de mogelijkheid om Euro-betaalpassen van een chip te voorzien. Naar de centrale kredietcommissie van de Westduitse kredietverlenende instellingen meedeelde zou dat de toepassingsmogelijkheden van de pasjes aanzienlijk verruimen.

Het ligt in de bedoeling bepaalde gegevens over de houders van de pas op te slaan in het geheugen van een chip die in de betaalpas wordt aangebracht. Daardoor wordt het niet alleen mogelijk contant geld te innen bij automaten maar bijvoorbeeld ook om inkopen te doen zonder geld of betaalcheques op zak te hebben.

Een winkelkassa, in feite een computerterminal, onderzoekt of de houder van de kaart wel voldoende geld op zijn rekening heeft staan. Is dit het geval dan schrijft de computer het bedrag van de rekening af. Ook het telefoneren uit een cel door het pasje in een daarvoor bestemde gleuf te duwen behoort in principe tot de mogelijkheden.

De kredietcommissie vindt dat de kaart in zoveel mogelijk landen gebruikt moet kunnen worden. Om dat doel te verwezenlijken hebben banken en kredietinstellingen in België, Frankrijk, Groot-Brittannië, Spanje, Ierland, Luxemburg, West-Duitsland en Nederland zich aaneengesloten in een Internationale Organisatie voor Microcircuit-kaarten.

Het aanbod in de IBM System 38 serie is weer uitgebreid:

IBM brengt Model 4 in Systeem 38 serie

De IBM System 38 serie is uitgebreid met het Model 4. Het nieuwe werkstationsysteem heeft een hoofdgeheugen van 512 tot maximaal 2048 kilobytes met een geheugencyclus van 200 nanoseconden. Het in stappen uit te breiden schijfgeheugen heeft een capaciteit van 64,5 megabytes tot 2,67 gigabytes. Maximaal tachtig werkstations kunnen worden gekoppeld in een lokale omgeving. Op afstand kunnen via de acht communicatielijnen onder het BSC- of SDLC-protocol enige honderden stations worden aangesloten. Dezelfde randapparatuur als voor de Modellen 3 en 5 is beschikbaar.

Computable, 7 augustus 1981

En reeds op 5 juni 1981 meldde Computable het verschijnen van een informatieboek over alles wat in Nederland met automatisering te maken heeft:

Computerjaarboek vanaf ESB 1981 vrij verkrijgbaar

Op de Europe Software Beurs toont de uitgever International Information Research haar boreling, het Automatiseringsjaarboek 81. Dit lijvige boekwerk met gedetailleerde gegevens over al het aanbod op de Nederlandse markt aan computerdienstverlening, -apparatuur en -toebehoren en programmatuurprodukten is daarna ook vrij verkrijgbaar. In gespecialiseerde boekwinkels en computershops zal het voor f 32,50 te koop zijn. Wil men het rechtstreeks van de uitgever betrekken, dan komt daar nog eens acht gulden porto bovenop. Hiervoor dient men zich te richten tot International Information Research, Postbus 6800, 6503 GH Nijmegen, dat telefonisch onder (080) 22 42 00 is te benaderen.

Beveiliging

De behoefte aan uitwijkcentra is wellicht even groot als de kosten die een dergelijke voorziening met zich mee brengt. Op 15 juni 1981 wijdt het Financieële Dagblad een artikel aan het initiatief van een drietal Burroughs-gebruikers die voor gezamenlijke rekening een uitwijkcentrum heeft opgezet. Uit dit artikel citeren wij het volgende :

Drie grote computergebruikers starten met gezamenlijk reserve-computersysteem

Automatiseren levert ook z'n risico's op. Ondernemingen en instellingen zijn vaak griezelig afhankelijk van het goed functioneren van hun computersysteem. Zij worden bedreigd door onoverzienbare rampen als hun computersysteem door brand, overstroming, sabotage en dergelijke in de vernieling gaat, en zij binnen enkele dagen niet kunnen voorzien in een operationeel reservecomputersysteem.

Hele grote computergebruikers kunnen het zich wellicht nog veroorloven binnen de eigen organisatie een volledig reserve-systeem te installeren. Maar dat is alleen weggelegd voor de allergrootsten.

Bij de iets minder grote ondernemingen en instellingen werken de kosten van zo'n eigen schaduw-systeem prohibitief. Een computersysteem van redelijke omvang is al duur genoeg. Een tweede reserve-systeem, dat pas echt nuttig wordt gebruikt als er een ramp in welke vorm dan ook, heeft plaatsgevonden, kan individueel niet worden opgebracht.

Van zo'n redenering worden de risico's echter niet kleiner. Naarmate het computergebruik oprukt, wordt de afhankelijkheid van de computer groter, en groeit het risico van de gevallen, waarin het computersysteem langdurig buiten gebruik raakt. Kortom dit vraagt om samenwerking en het delen van de risico's. Als één onderneming een reserve-computersysteem ("back-up"-computer) niet kan betalen, kunnen meer computergebruikers dat gezamenlijk wel. Vele gebruikers gezamenlijk kunnen volstaan met één en hetzelfde reserve-systeem, mits zij dezelfde soort computerapparatuur gebruiken. Het risico wordt dan gedeeld, evenals de kosten.

Dit is de gedachte achter de oprichting van Comsure BV in Rotterdam, een bedrijf dat zich vanaf 1 augustus aanstaande gaat specialiseren in het verlenen van zogenaamde "back-up"-faciliteiten bij het verloren gaan van de computerapparatuur van een onderneming door brand, sabotage, overstroming of andere vormen van calamiteiten.

Comsure is opgericht door drie grote computergebruikers : Europe Container Terminus ECT, een van de grootste West-europese containerbedrijven in Rotterdam, de Westland-Utrecht Hypotheekbank en de Combinatie Samenwerkende Rekencentra CSR, als computerservicebureau een divisie van Centraal Beheer. Deze drie ondernemingen, alle drie gebruikers van de middel-grote Burroughs B1985-computersystemen hebben in eerste instantie vooral zelf een zeer grote behoefte aan back-up-faciliteiten voor het geval dat onverhoopt hun eigen computers door een ramp worden vernietigd.

Kosten

De kosten, die de Westland-Utrecht, ECT en CSR hadden moeten maken voor een individueel reserve-systeem, worden dankzij de samenwerking door drie gedeeld. Dat levert niettemin nog een forse kostenpost op, waar de drie ondernemingen enigzins zorgelijk tegenaan kijken. De twee huidige computersystemen, die in Rotterdam worden geïnstalleerd, zijn van Burroughs geleased voor een periode van twee jaar. Dat kost iets meer dan f. 3 mln ofwel f. 1 mln per partner over twee jaar. Met andere woorden : in deze opzet heeft ieder van de drie partners zich verzekerd tegen computercalamiteiten à raison van f. 500.000,-- per jaar. Voorwaar een forse premie. En dan is dit bedrag nog laag omdat Comsure BV onderdak heeft gevonden in een bestaande behuizing van CSR in het Rotterdamse Groothandelsgebouw, en omdat er een beroep kan worden gedaan op de aanwezige systeem- en softwarekennis van CSR. Toch vinden ook de drie samenwerkende partners in Comsure BV de premie voor hun risicodekking aan de hoge kant. Daarom gaat Comsure BV proberen andere Burroughs-gebruikers te interesseren voor de in Rotterdam aanwezige back-up-faciliteiten. Ondernemingen met Burroughs-computers in huis, die zich net als de drie oorspronkelijke partners bewust zijn van de risico's van calamiteiten, kunnen zich bij Comsure BV verzekeren van capaciteit op een vergelijkbaar computersysteem voor als in eigen huis de nood aan de man is. Dit recht op uitwijk op de B1900-configuratie bij Comsure kan worden verkregen tegen een gemiddelde jaarpremie van ongeveer f. 15.000,--. Voor het kunnen uitwijken naar de B3900-configuratie moet een jaarpremie tussen de f. 25.000,-- en f. 50.000,-- worden betaald, al naar gelang het feit of de potentiële gebruiker aan de onderkant of aan de bovenkant van dit Burroughs-systeem zit.

Het initiatief voor Comsure BV komt voort uit een drie jaar geleden opgerichte werkgroep van de Burroughs-gebruikersclub, die de risico's van computercalamiteiten ging bestuderen. Deze werkgroep organiseerde een enquête onder enige honderden Burroughs-gebruikers.

Overleven

Eén van de vragen was : Hoe lang kan een onderneming het verloren gaan van de computerfaciliteiten overleven? De antwoorden stemden niet vrolijk. Ruwweg 23% van de ondernemingen zei het niet langer dan één dag zonder computer te kunnen stellen, wilde men niet in ernstige problemen komen. Zo'n 33% van de bedrijven verklaarde het twee dagen te kunnen volhouden zonder ernstige problemen, en zo'n 13% hield het op drie dagen. Na enige weken van computeruitval zou de continuïteit van de hele onderneming zelfs in gevaar komen, zo vreesden vele bedrijven.

Nu, met de on-line systemen, zijn veel bedrijven met huid en haar opgehangen aan het functioneren van het computersysteem. Met alle rampzalige gevolgen van dien, als die faciliteit plotseling wegvalt.

Waar vind je zo gauw een beschikbaar computersysteem, waarbinnen op zo'n 100 terminals aan dagelijkse up-dating kan worden gedaan.

Zelfs Burroughs als onze leverancier is niet in staat ons dit te garanderen. In contracten van vijf jaar geleden, stelt Burroughs zich wel verantwoordelijk voor het verlenen van back-up-faciliteiten in het geval van calamiteiten. Maar Burroughs geeft zelf toe, dat ze dat niet meer waar kunnen maken, dat ze die toezegging niet meer kunnen nakomen.

Dat de wet op de privacybescherming te lang op zich laat wachten is een punt waar velen het mee eens zijn.

Dat er hier en daar al vast op wordt geanticipeerd zien we helaas maar weinig. Nieuws vormt dan ook onderstaand bericht uit Adformatie van 14 mei 1981 :

Notu komt aan wet tegemoet met privacy-reglement

De leden van de Notu hebben een modelreglement goedgekeurd voor het beheer van persoonsgegevens. Daarmee lopen de tijdschriftuitgevers vooruit op de komende wet ter bescherming van de persoonlijke levenssfeer.

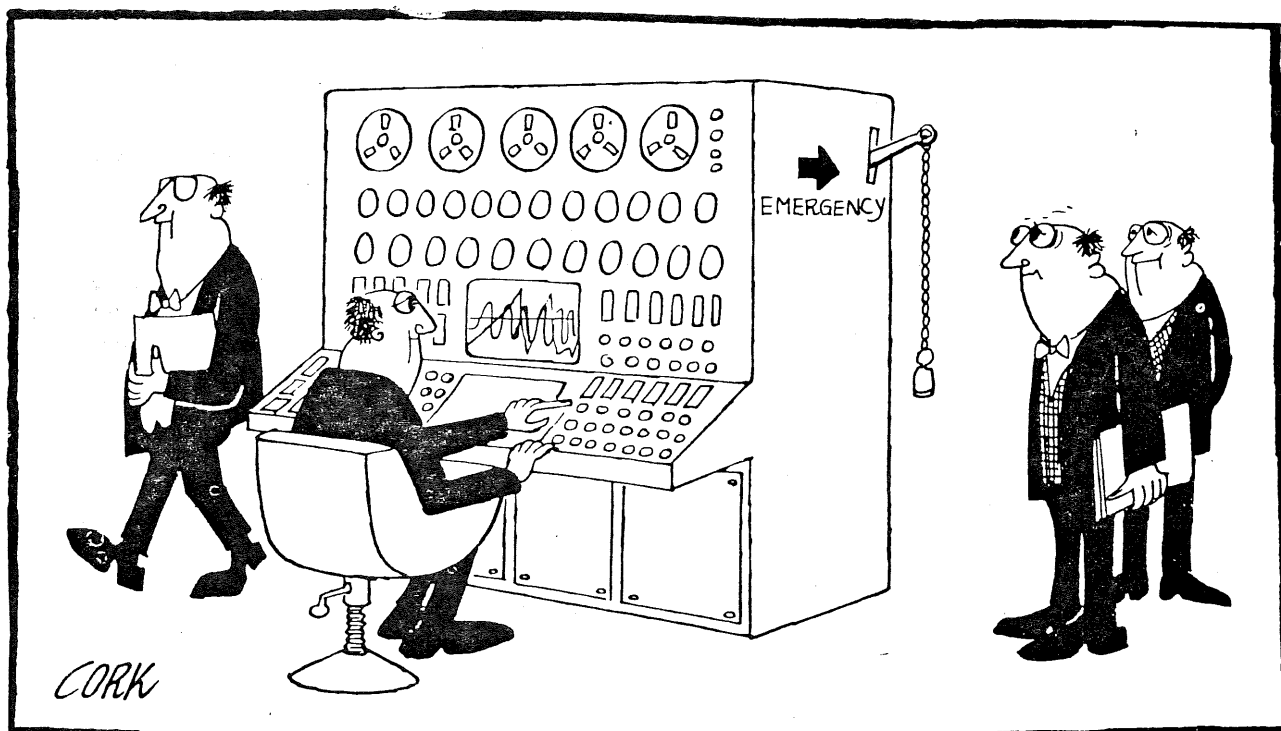
Met het modelreglement wil de Notu haar leden een hoop werk besparen. Dat kan, als de Registratiekamer waarin de nieuwe wet voorziet, het Notureglement aanvaardt als standaard. Het modelreglement en de Notu moet voorkomen dat meer persoonsgegevens worden verwerkt dan nodig voor het gestelde doel, in artikel twee omschreven als "een verantwoorde uitoefening van de administratieve en commerciële functie van de uitgeverij".

Het eerste artikel in het reglement noemt een commissie van toezicht met drie onafhankelijke deskundigen, die toezicht moet houden op naleving.

Op de terbeschikkingstelling aan derden van adresbestanden gaat het reglement onder meer ook in. Adressen van personen die daartegen bij de Notu schriftelijk bezwaar hebben gemaakt, gaan niet naar derden toe tenzij de wet de uitgever daartoe verplicht.

Als iemand uit het bestand wil omdat hij geen commerciële mededelingen en aanbiedingen meer wenst te ontvangen, moet de bestandhouder er voor zorgen dat het schrappen binnen uiterlijk drie maanden zonder enige restrictie gebeurt. Heeft iemand een dergelijk verzoek tot schrapping gedaan aan de Notu of rechtstreeks aan de houder van het adresbestand, dan zal de bestandhouder hem schriftelijk laten weten dat het ook is gebeurd.

Tenslotte een cartoon van CORK die wij aantreffen in
Der Datenschutzberater



THE COMPARATOR : A NEW AUDIT TOOL

Aldus de kop van een artikel in EDPACS. Of deze audit tool zo nieuw is moet echter worden betwijfeld. Een comparator wordt gedefiniëerd als : "Een instrument dat geautomatiseerd data vergelijkt op lengte, afstand, kleur, etc."

Van belang voor de accountant echter is een computerprogramma dat twee versies van programma's in sourcecode vergelijkt en de wijzigingen ten opzichte van het origineel aangeeft. De aangebrachte wijzigingen worden op een overzicht afgedrukt.

In het volgende artikel geeft J. Yingling een beschouwing over welke mogelijkheden een comparator package voor de accountant zou moeten hebben.

Van elke mogelijkheid wordt aangegeven wat deze inhoudt en welk nut het kan hebben voor de accountant. Tot slot gaat de schrijver summier in op een praktijkgeval.

Ter aanvulling op het artikel is informatie opgenomen over twee "comparators" :

- Source comparison utility;
- New source code compare program.

Who cares about program changes?

Most audit publications have had at least one article about the weakness of controls over changes to programs. The changes cannot be easily reviewed to determine that they were properly authorized. The problem has been that there was no easy way to isolate changes from the unchanged code.

In-house changes to purchased software packages are particularly troublesome. Vendors usually send out a completely new version of the package at least once a year. Programmeurs cannot be assumed to have marked all "in-house" changes so that they can be easily identified and incorporated into the new version of the software. A comparator can be used to identify those changes by matching the original source code supplied by the vendor with the current, "in-house" version.

What to look for in a comparator

A comparator package should provide the following facilities :

1. Compare two source code versions of a program;
2. Compare two load module (object code) versions of a program;
3. Compare two library data sets, either source or load module form;
4. Compare two print-image data sets;

5. Produce a file of changes following each update and have the capability of using that file to :
 - Print a report of changes
 - Convert a "new" version of a program into its corresponding "old" version.
6. Be easy to install and operate. Instructions should include examples of the job control language (JCL) required for the common functions of the comparator.

Source code comparisons

The capability to compare two versions of program source code allows the auditor to independently verify that program changes were properly authorized. A high level language, such as COBOL, should be comprehensible to most EDP-auditors. If the auditor is reviewing changes to the operating system, however, he may need assistance from a systems programmer or an outside consultant. The auditor needs assurance that the source code being examined produced the executable version in the library. This is more likely if an independent party, such as a librarian, assumes responsibility for accepting source code, compiling it into production programs, and recording the source code on a production library.

Another means of such assurance is to compile the source code for the auditor and compare the object code to the version in the production library.

Load module comparisons

This facility allows the auditor to verify that production programs have not been subject to unauthorized changes. The auditor obtains a copy of the load module after an authorized change and keeps it under his control until it is needed for comparison with the current production version.

Library data set comparisons

The facility for comparison of a data set allows the auditor to compare entire program libraries without having to name the programs they contain. Such comparisons are particularly useful in auditing "in-house" source code changes which have been applied to packages obtained from outside vendors. To facilitate this control approach, all source code versions of purchased software should be stored on the same data set.

Even installations that are using RACF or ACF2 to protect their libraries will find that library comparison is a useful function. Systems programmers, particularly those that maintain the software for the security system, can circumvent the controls provided by RACF, ACF2, and similar software. Library comparison can help reduce this risk. In effect, the security software can be viewed

as a lock and the library comparison can be equated to a physical inventory of data content. Used together, these facilities represent a comprehensive audit and security program.

Print-image comparisons

Programmers can use print-image comparisons during parallel testing when the printer output is not expected to undergo any major changes. Output from both the "old" and the "new" systems is directed to disk or tape and then compared. Results from the comparison are reviewed to determine if any differences beyond those that were expected were generated. Use of this comparison technique can be an invaluable aid in a conversion.

Change files

Auditors can use the change files to maintain a history of program modifications. Further, the auditor or users can utilize these change files to work backwards from the current version of a program's source code to economically reconstruct prior versions, rather than retaining the original copies. These facilities are useful in maintaining control over program changes.

Installation and operation

To be most useful for the auditor, the comparator should be a "load and go" software package. There should be no need to compile, generate or install the system each time it is used.

Practical uses of comparators

The posting portion of a 15-year-old demand deposit system was converted from IBM assembly language to VS/COBOL. A few system changes were involved, but for the most part the new system was expected to produce the same printed reports as the old system. The data processing staff used the comparator to match 250,000 print lines from each system to detect any unanticipated differences. A few differences came to light, but they were not considered to be serious. Since the converted system has gone live, no subsequent conversion-related problems have been encountered.

In the same bank installation, a comparator was used to match an original copy of IBM's Check Processing Control System (CPCS) against the modified version used in normal production processing. The modifications that were detected were then applied to the new version of CPCS.

On a daily basis, the comparator is used to compare versions of program source code. In particular, such comparisons are made when an error in testing is encountered. Such errors can usually be traced back to the portion of code that was changed.

Conclusion

Comparators can be an extremely powerful multifunctional tool for auditors, security professionals, and the data processing staff. Care must be exercised in selecting a package and planning its use.

EDPACS
april 1981

Source comparison utility

A program product, SRCCOMPR, is available for use in comparing two files of program source code. The package provides the following features :

1. Comparison on specified columns within the input records. For example, COBOL programs might be compared on positions 8 through 72 in each record.
2. Detection of inserted or deleted records. Vendor supplied utilities cannot perform this function. When SRCCOMPR (referentie 1) detects a mismatch it begins to search for the next match between the files and isolates the sections of mismatched code.
3. User control over matched line count. A pair of matched lines, as noted in section 2 above, terminates the print-out. This may not satisfy the user's requirements. From the standpoint of meaningful analysis, it might be more significant to have a report tailored to the user's requirements so that large blocks of code can be printed out even though some of the lines of code may match.
4. Ignore blanks during record comparisons. The package can be instructed to skip blank characters when comparing lines of program code.
5. Ignore program comments. Comment lines in languages such as FORTRAN, COBOL and RPG, can be ignored when making comparisons.

While it has obvious application during operating system conversions, this package is also useful for maintenance operations and auditing. In particular, it can be used to highlight changes within a program. A file of SRCCOMPR listings provides an audit trail for the review of program maintenance activity. Further, the package can be used to compare data files and JCL.

EDPACS
juni 1981

1) SRCCOMPR is een produkt van MIS Associates.

New Source Code Compare Program

This program, called Test Comparator and for use on IBM S/370 systems, can be used to compare two versions of a source program, or similar data sets, without regard for sequence or line numbers. Changes between the two versions being compared are reported and provide an automated audit trail of all revisions. Since most source code compare software available up to now makes comparisons based upon line numbers, this product represents a new level of sophistication. A 90-day free trail is offered.

De Test Comparator is een produkt van Yingling Associates, Inc.

EDPACS
april 1981

In EDPACS van maart 1981 lazen we onderstaande ervaringen van een RACF-implementatie door een EDP-audit manager. Vooral de inhoud van de laatste alinea doet merkwaardig aan gezien het feit dat ze komen uit de pen van een EDP-audit manager.

RACF implementation successful

I am the EDP Audit Manager for a very large manufacturing company which uses RACF. Security software was implemented by MIS management because they had to protect the information assets for which they were directly responsible (system data sets, program libraries, proprietary packages) and those for which they were custodians (application files and data bases).

We have a very simple policy on computer-based information security :

1. Information that is "critical" to the operation of the company (e.g., order entry, inventory management system) must be protected against unauthorized change (write protection) and must have adequate backup.
2. Information that is "private" (e.g., salaried payroll, human resources systems) must have both read and write protection.
3. Information that could be used to perpetrate a fraud (e.g., payables, receivables, inventory systems) must have write protection.

The user is responsible for classifying information. MIS is responsible for providing the required protection, and the access control software tool being furnished is RACF. RACF was selected in order to ensure operating system compatibility and product support and to avoid proliferating a multi-vendor environment. Furthermore it meets our security goal of protecting only what needs protection, not everything.

I focus on auditing users, not MIS, for compliance with the policy. RACF works and has been implemented for all asset-related applications in one data center, including VSAM data bases accessed by IMS. Another data center is converting from DOS to MVS, and RACF will be implemented as part of the conversion.

RACF protects only what we specifically wanted protected.

One fulltime person administers security.

The concerns I have about RACF are related to its very weak auditor capability and the problems of passing RACF-protected tapes between separate systems. However, these concerns are minor and will be addressed as the product matures. RACF has been easy to install, implement, and administer.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.

EXTERNE CURSUSSEN

ONDERWIJS

=====

door H. Lafèbre

Kleinschalige automatisering

Door de AC groep is in samenwerking met de Organisatiegroep een nieuwe 3-daagse cursus op het gebied van de kleinschalige automatisering (mini's en micro's) ontwikkeld. De ontwikkeling heeft plaats gevonden in opdracht van de Commissie Voortgezette Educatie Registeraccountants (VERA) van het Nederlands Instituut van Registeraccountants (N.I.v.R.A.). De cursus is opgenomen in het interne opleidingsprogramma van KKC.

Verwacht kan worden dat de cursus in 1982 ook voor medewerkers van onze cliënten zal worden opengesteld.

De cursus is bestemd voor registeraccountants en ervaren controleleiders in de algemene controlepraktijk, die over een basiskennis ten aanzien van automatisering beschikken en die als adviseur en controleur werkzaam zijn in kleine en middelgrote bedrijven waar mini- of microcomputers zijn of worden geïnstalleerd.

Tijdens de cursus wordt beoogd aan de accountant als adviseur van de bedrijfsleiding van kleine en middelgrote bedrijven inzicht te geven in de wijze waarop een bedrijf de automatisering moet aanpakken. Tevens wordt aan de accountant als controleur kennis meegegeven van een aantal belangrijke organisatorische en technische aspecten, die van belang zijn voor de wijze waarop hij het beste het bedrijf kan controleren.

Vóór de eerste cursusdag wordt aan de deelnemers een niet te uitgebreide syllabus toegestuurd. De belangrijkste gedeelten hieruit worden toegelicht in korte inleidingen. Aan de hand van zes cases -die alle betrekking hebben op één bedrijf- wordt de stof praktisch uitgewerkt. Voor later gebruik worden lijsten met attentiepunten voor de verschillende problemen verstrekt.

De syllabus heeft betrekking op de onderwerpen:

- . een aantal meer technische aspecten van de mini- en micro-computer;
- . de organisatie van de automatisering;
- . de accountantscontrole bij geautomatiseerde gegevensverwerking.

Het inzicht in de automatiseringsaanpak wordt verkregen met behulp van inleidingen en cases over:

- . de beoordeling van het automatiseringsplan (keuze eigen computer of uitbesteden bij een servicebureau);
- . de bepaling van de informatiebehoeften;
- . de vergelijking van twee offertes van computerconfiguraties.

In de daarop volgende twee cases worden de interne controle-aspecten van een aan te schaffen (niet standaard) programmapakket voor de orderverwerking, facturering en voorraadadministratie en van een standaardprogrammapakket voor de financiële administratie beoordeeld. Tenslotte worden in de laatste case de externe controle-aspecten behandeld, welke samenhangen met de verwerking van de te controleren gegevens door een mini- c.q. microcomputer.



COMPACT is een uitgave van de AC-groep van Klynveld Kraayenhof & Co.