

„It would be an inconvenient rule if nothing could be done until everything can be done.”

Churchill

I INLEIDING

a. Het ontwerpen van informatiesystemen voor automatische verwerking met rekentuig is geen sinecure. De meest uitgebreide studie hiervoor dreigt echter nutteloos te worden indien bij de gegevensinvoer fouten worden gemaakt

Met voorbijgaan aan de zeer omvangrijke problematiek, verbonden aan het systeemontwerp als zodanig, willen wij in het kader van dit artikel nader ingaan op een aantal aspecten van de organisatie van de invoer, een onderwerp dat in vakkringen nog weinig aandacht heeft.

b. Allereerst een aantal begrippen. Hoewel wij in het vervolg van dit artikel zullen trachten nederlandse termen te gebruiken, is het - helaas! - voor de duidelijkheid noodzakelijk, de frekwent gebruikte engelse equivalenten als toelichting te vermelden.

Bestand (= file)

Een bestand is een samenhangende verzameling gegevens inzake een informatie-objekt, als regel opgebouwd uit een aantal bestanddelen.¹⁾ Voor hen die de term „bestand” als een germanisme afwijzen, kan de term „lias” worden aanbevolen. Ook kan nog genoemd worden het door Prof. Starreveld geïntroduceerde „informatieverzameling”: Wij menen echter dat dit woord te lang is voor dagelijks gebruik.

Bestanddeel (= logical record²⁾)

Een bestanddeel is een samenhangende verzameling gegevens inzake een component van een bestand en is als regel opgebouwd uit een aantal velden.³⁾

Het komt veelvuldig voor, dat een bestanddeel een variërend aantal velden bevat.

Alternatieven voor de term bestanddeel zijn: onderdeel, rekening, post, (boekings-)hoofd, element.

1) Honeywell Glossary of Data Processing and Communications terms: „File = a collection of related records”.

2) Bepaalde auteurs, bijv. Laden & Gildersleeve in „System design for computer applications” gebruiken „record” en „item” door elkaar; elders, bijv. in Cobol-61, wordt „item” in plaats van „field” gebruikt als onderverdeling van „record”.

3) Honeywell Glossary: „Record = a group of related facts or fields of information treated as a unit, . . .”.

Veld (= field)

Een veld is een gegeven dat een bepaalde, eigen, betekenis heeft en is opgebouwd uit een aantal tekens (= characters).⁴⁾ ⁵⁾

Deze term wordt ook gebruikt voor de technische presentatie van het gegeven: vergelijk bijv. de uitdrukking „kaartveld”.

Een veld heeft veelal een vaste lengte; bij ponsbandinvoer kan hiervan ter besparing van inleestijd op eenvoudige wijze worden afgeweken.

De definiëring van de reeks termen „bestand - bestanddeel - veld” is gebaseerd op de informatie-*inhoud*.

Hiernaast komen termen voor, die betrekking hebben op de verschijnings*vorm* van de informatie:

Blok (= block)

Een blok bestaat uit een aantal gegevens, dat bij in- en uitvoer van rekentuig als één geheel wordt behandeld; het is als regel opgebouwd uit een aantal subblokken.

Subblok (= physical record)

Een subblok is een komponent van een blok en bestaat uit een aantal gegevens, als regel inzake een verrichte bedrijfshandeling of inzake een bestand en is opgebouwd uit een aantal velden.

Bijv.: een ponskaart of door sluittekens afgebakende groepen van velden op een ponsband of magneetband.

Het aantal velden hoeft niet gelijk te zijn aan de groep velden, die tezamen een bestanddeel of een volledige mutatie vormen.

Ter verduidelijking een schematisch weergave:

BESTAND					
BESTANDEEL					
		VELD			
BLOK					
	SUB-BLOK				

II INDELING VAN KONTROLE-KATEGORIEËN

a. *Technische of ingebouwde controles*

Voor de vele elementaire, technische kontrolemogelijkheden die los staan van de informatiesystemen maar veelal als standaard zijn ingebouwd in apparatuur (=

4) Honeywell Glossary: „Field = a specified area of a record used for a particular category of data . . .”.

5) IBM, manual Decision tables: „A field contains a value or unit of information”.

hardware) of in fabrikantenprogramma's (= software, zoals sorteren, assembleren met bijv. makro-opdrachten voor gegevensin- en uitvoer van het rekentuin) wordt verwezen naar het boek van Van Belkum en Van 't Klooster.⁶⁾ Deze ingebouwde controles dienen voornamelijk als beveiliging tegen verminking bij de informatiebehandeling door machines en worden dus beheerst door de stand van de techniek.

b. *Produktiebeveiliging*

Evenmin wil ik in het bestek van dit artikel op een andere reeks van controlemaatregelen ingaan, die bijv. betrekking hebben op beveiliging van het bewaren van programma's en bestanden en op het tegengaan van akkurateisfouten van operateurs. Deze controles hebben hun plaats binnen de afdeling voor machinale verwerking.

c. *Procedure- of Invoercontroles*

De controle op de invoer omvat het traject vanaf creatie bij de bron van de in te voeren gegevens tot en met de overdracht voor machinale verwerking.

Gezien de veelal bestaande funktiescheiding en de geografische afstand is dit een gebied waar controles niet alleen mogelijk, maar ook zeer noodzakelijk zijn.

d. *Te programmeren of systeem-controles*

Speciaal een informatiesysteem dat een omvangrijke integratie in de gegevensverwerking heeft verankerd in een samenstel van computerprogramma's is van grote betekenis voor controlehandelingen.

Niet alleen omdat meerdere controlesoorten doelmatig kunnen worden toegepast, maar tevens omdat de programma's waarin de controlestappen zijn opgenomen een betere garantie geven voor daadwerkelijke en konsekvente toepassing dan een funktionaris die hiervoor verantwoordelijk zou zijn.

Vele van de te programmeren controles zijn controles op de invoer en in zoverre is de benaming van punt c hierboven niet juist. Dit vindt zijn oorzaak in het feit dat bij volgtijdige verwerking (serial of sequential processing) de invoer eerst tot partijen wordt geformeerd voordat tot de eigenlijke verwerking wordt overgegaan. In deze voorfase is het efficiënt, tevens reeds zo ver mogelijk de invoer te zuiveren (screenen).

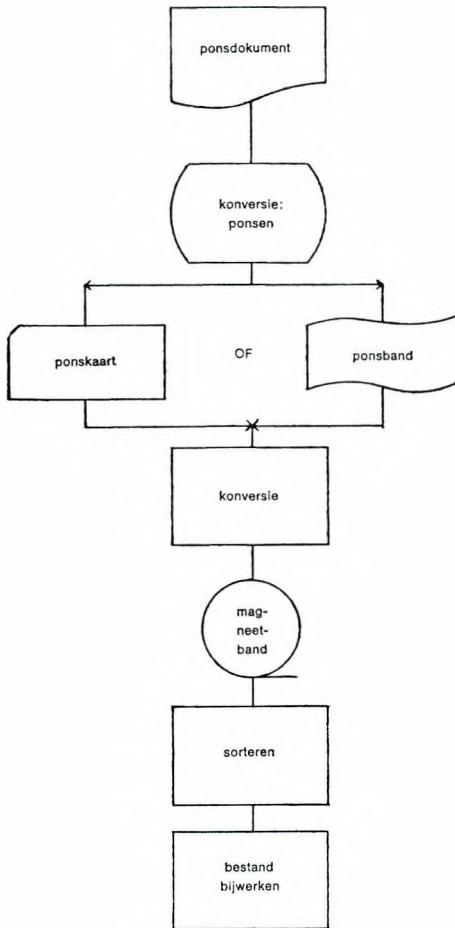
Wij hebben in deze indeling de term organisatorische controle vermeden omdat wij deze te ruim achten: De categorieën b t/m d kunnen hiertoe gerekend worden.

In het nu volgende zullen wij met voorbijgaan van de controleaspecten die met de eigenlijke werking van het systeem samenhangen ons beperken tot de mogelijkheden tot beheersing van de invoer.

6) J. W. van Belkum en A. J. van 't Klooster, „Administratieve Automatisering en Controle”, Alphen aan de Rijn 1964.

III OPZET VAN VERWERKING EN BESTANDINDELING

a. De gegevensinvoer doorloopt als regel meerdere fasen. Een gebruikelijk schema in geval van partijgewijze verwerking (= batch processing) is het volgende:



Van belang hierbij is de volgorde (= sequence).

Een bestand heeft veelal een konstante volgorde; in het algemeen geldt dat de volgorde, die een minimum aan sortering voor in- en uitvoer vereist, de beste is.

Gezien de veelal naar verhouding grote omvang dient het omsorteren van bestanden vermeden te worden: Hieruit volgt dat de gegevens die bepalend zijn voor de volgorde een zo vast mogelijke waarde moeten hebben.

De sorteervolgorde wordt bepaald door het *sorteerkenmerk* (= key). Dit bestaat uit één of meer tekens c.q. één of meer velden die in de regel de identifikatie van het bestanddeel of van het „physical record” vormen.

b. Voorbeeld:

In een bestand, dat is opgebouwd uit bestanddelen van verschillende lengte is een hoofdsortering per bestanddeel en een subsortering per „physical record” aangebracht.

Bij de zgn. permanente velden zal tenminste een veld aanwezig zijn dat het hoofdsorteermerk bevat alsmede een identifikatiemogelijkheid van het permanente veld. In de tekening is dit door een ster aangegeven. Bij de zgn. fakultatieve velden daarentegen slechts een veld voor het subsorteerkenmerk:

subblok

1 2 1 4 5	permanent	*
4	fakultatief	
6	fakultatief	
1 2 4 1 6	permanent	*
5	fakultatief	

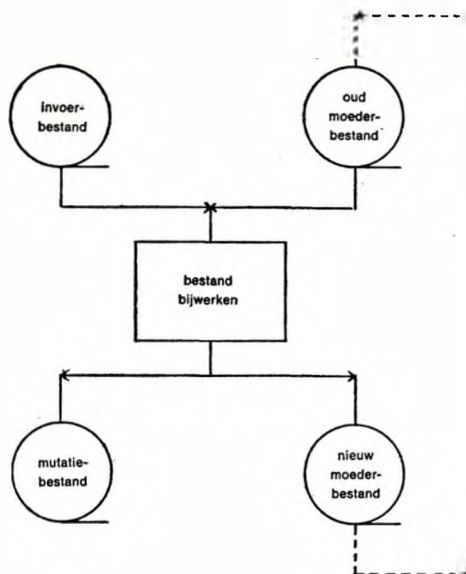
Indien in het systeem het omsorteren van het bestand noodzakelijk is verdient het aanbeveling, de sorteerkennmerken in elk subblok compleet op te nemen, hoewel dit uiteraard tot bestandvergroting leidt:

1 2 1 4 5	0	permanent
1 2 1 4 5	4	fakultatief
1 2 1 4 5	6	fakultatief
1 2 4 1 6	0	permanent
1 2 4 1 6	5	fakultatief

c. De moderne technieken maken het mogelijk essentiële informatie in logische samenhang en in één vorm vastgelegd snel machinaal toegankelijk te maken.

Efficiëncy-stelregels als „vermijd dublures” en „combineer gelijksoortige handelingen” vinden hun weerspiegeling in systeemopvattingen als de zgn. „integrated” of „consolidated file approach”.

Wanneer dus in het vervolg kortweg over „bestand” gesproken wordt, stellen wij ons een dergelijk moederbestand (= master file) voor ogen. Daarnaast is het uiteraard toegelaten, elke andere gegevensverzameling, zoals een partij nieuwe invoer of een over een periode gekumuleerde hoeveelheid - verwerkte - mutaties een mutatiebestand (= transaction file) te noemen.



IV INVOERSOORTEN

Een bestand wordt bijgewerkt op grond van invoer. Wij onderscheiden twee hoofdsoorten invoer:

- bestandwijzigingen (= file alterations)
- mutaties (= transactions).

Bestandwijzigingen kunnen worden onderverdeeld in:

- toevoeging van bestanddelen (= additions)
- lichten van bestanddelen (= deletions)
- wijziging van sorteerkennmerk (= transfers)
- wijziging van inhoud van permanente velden (= changes)

Mutaties kunnen worden onderverdeeld in:

- toevoeging van fakultatieve velden
- lichten van fakultatieve velden
- wijziging van inhoud van permanente velden
- wijziging van inhoud van fakultatieve velden

In de praktijk komen combinaties van deze invoersoorten frekwent voor.

Ondanks het onderscheid dat in de invoer en de gevolgen ervan kan worden gemaakt zijn geen principiële redenen voor afzonderlijke verwerking te geven.

Gezamenlijke invoer in één machinegang bevordert de uniformiteit van het systeem en reduceert de verwerkingstijd.

Aan dit streven worden grenzen gesteld door te complex wordende veldindeling van de invoer en te omvangrijke programma's.

V INVOERVERWERKING: HET PONSDOKUMENT

a. In deze paragraaf wordt uitgegaan van het reeds aanwezig zijn van invoer bij

de ontstaansbron (source documents) en dus niet gesproken over - visuele - juistheidkontroles bij het aanmaken van invoerdokumenten c.q. bij het overnemen op speciale ponsconcepten.

Ook gaan wij hier voorbij aan de verdiensten van zgn. uitlichtmethoden van voorgeponste en vertolkte ponskaarten, aangezien deze een afwijkende procedure hebben. Er moet dan gezorgd worden voor het op peil houden van het uitlichtbestand en er moet gelicht worden. Deze extra handelingen vervangen geheel of gedeeltelijk het ponswerk waardoor een aantal van de hierna te bespreken controles niet mogelijk is. Daar staat tegenover, dat er slechts beperkte - visuele - controles mogelijk zijn die moeten waken tegen de kansen op foutief lichten en tegen verwisseling van kaarten bij het ponsen van variabele gegevens.

b. Het bestek van een tijdschriftartikel leent zich niet voor een gedetailleerde opsomming van de verschillende handelingen die de invoer voor een automatisatiesysteem moet ondergaan. Wel zullen wij trachten een beeld te geven van de aard van de controles die op dit terrein aan de orde kunnen worden gesteld.

Wanneer men de verwerking van het ponsdokument beziet vanuit controle-oogpunt dan valt de nadruk op akkurateesse- en volledigheidskontroles of combinaties van beide. Nagegaan moet worden of het dokument volledig is ingevuld; andere voorschriften kunnen betrekking hebben op het voortellen per dokument of per partij documenten, op het registreren van gekreëerde documenten of op het vermelden van vaste controletekens of controlegetallen.

Speciaal bij sorteergegevens is een extra waarborg voor juiste invoer zeer gewenst. Een voorbeeld hiervan is, dat aan kodenummers een controleteken wordt toegevoegd dat vooraf bepaald werd op grond van een rekenkundige bewerking met de cijfers van het kodenummer.

Bij het ponsen wordt het controleteken onmiddellijk na het kodenummer aangeslagen; de „check digit verifier” van de ponsmachine berekent het controleteken op grond van het geponste kodenummer en vergelijkt zijn uitkomst met het aangeslagen controleteken; bij verschil wordt de machine geblokkeerd.⁷⁾

Als organisatorische maatregel is het ter versteviging van de controle op volledigheid en voortgang gewenst dat de hierbedoelde handelingen zoveel mogelijk bij de bron van informatie geschieden.

VI KONVERSIE NAAR PONSKAART OP PONS BAND

a. De konversie is in principe omzetting van het ponsdokument in karakterschrift naar een informatiedrager in positieschrift.⁸⁾ Dat deze informatie-drager tevens een vertolking in karakterschrift kan bevatten of dat tegelijkertijd een aparte publikatie (= report) wordt geproduceerd is bijzaak.

De nieuwe informatiedrager is door zijn strikt gedefinieerde indeling een subblok of physical record.

7) Zie voor een voorbeeld de casus in het Rapport subcommissie C.A.B.: De Accountant april 1965, blz. 499 - 531.

8) Wij ontleen deze termen, die kernachtiger zijn dan resp. „menselijk leesbaar schrift” en „machinaal leesbaar schrift” aan de inaugurele rede van Prof. J. Nathans „De administratieve informatieverzorging met betrekking tot de beslissingsvoorbereiding op korte termijn”; Amsterdam, 1964. Als een taalkundig juister alternatief voor karakterschrift zij hier nog melding gemaakt van „letterschrift”.

In dit stadium kunnen zowel handmatige als gemechaniseerde controles toegepast worden. Handmatige zoals het uit een oogpunt van volledigheidskontrolé vastleggen in een register. Gemechaniseerde controles enerzijds door het dubbelrend gebruiken van de apparatuur bijv. bij controleponsen (een handeling die „verification” wordt genoemd) en anderzijds door gebruikmaking van additionele mogelijkheden van de konversiemachine bijv. het afdrukken van de reeds genoemde publikatie, het narekenen van controletekens en controlegetallen, het programmatisch nagaan van de formele juistheid van veldlengte, of van invulling en gebruik van toegestane alfabetische, numerieke, alfanumerieke of speciale tekens. Deze laatste handelingen worden tot de „validation” gerekend.

Zowel de „verification” als de „validation” vallen o.i. onder het begrip akkurateessekontrolé. Met dien verstande, dat de publikatie zowel als het natellen van controletotalen per partij dokumenten tevens een middel tot volledigheidskontrolé verschaffen.

Een - machinale - kontrolé op het gebruik van geldige tekens in bepaalde velden resp. op het gebruik van tekens van een bepaalde waarde reikt verder dan het wezen van de akkurateessekontrolé. In zo'n geval kan beter van juistheids- of geldigheidskontrolé gesproken worden.

Het behoeft geen betoog, dat vele van de hierbedoelde controles simultaan kunnen worden toegepast.

Een zeer belangrijk voordeel van zo grondig mogelijke kontrolé bij de eerste konversie van de invoer is dat de terugkoppeling naar de bron maximaal is qua foutdetectie en minimaal qua tijdverlies. Dit pleit tevens voor de wenselijkheid van decentrale invoerkonversie ook bij gebruik van centraal in het bedrijf opgesteld rekentuiq.

De rekengangen (= runs) op dit rekentuiq behoeven dan niet of minder te worden onderbroken c.q. voorafgegaan te worden door een controlelgang.

b. Fouten in voortellingen

Het blijkt dat in de praktijk telfouten-in-voortellingen frekwenter voorkomen dan ponsfouten. Dat dit een ernstig bezwaar vormt van deze kontrolémethodiek wordt nog geaksentueerd door het feit, dat het de aandacht voor een goede invoerbehandeling eer doet verslappen dan intensiveren.

Het afhandelen van telfouten kan op twee plaatsen gelegd worden: bij de bron of bij de konversie.

Wanneer bij de nulproef op grond van één van de konversiehandelingen een afwijking aan het licht komt is het gebruikelijk dat ter plaatse de oorzaak wordt opgespoord. Blijkt de oorzaak een ponsfout te zijn, dan dient men zelf een korrektiehandeling te verrichten. Blijkt de oorzaak een telfout te zijn, dan kan men de voortelling verbeteren, dit aantekenen in bijv. een ponsregister of op andere wijze aan de informatiebron signaleren en vervolgens de korrektiehandeling uitvoeren. Dit bevordert de snelheid van verwerking en het bijhouden van invoerpartijen. Alleen in geval van twijfel, bijv. bij de interpretatie van slecht leesbare gegevens, of in geval van vereiste absolute funktiescheiding wordt deze invoer als uitschieter beschouwd.

c. *Uitschieters*⁹⁾

Uitschieters zijn hier ponsdokumenten die niet verwerkt kunnen worden. Dit betekent in de praktijk dat de documenten retour gaan van bijv. ponskamer naar lijnafdeling. Alleen al gezien de vertraging die dit veroorzaakt zijn wij van mening dat men hiertoe niet snel moet overgaan. De retournering gaat gepaard met vermelding van de reden van uitschieten.

Aangezien uitschieters de volledigheidskontrolle kunnen ondergraven, dient zorg te worden besteed aan:

- verlagen van partijgewijze voortelling en afstemmen met de bij konversie gemaakte natelling.
- uitschieters voorzien van oorspronkelijk partijnummer en het verlagen van stuktal op geleidebon en in kontroleregister indien de veronderstelde procedures bij de invoer worden gevolgd.

VII KONVERSIE NAAR MAGNEETBAND

a. Bij konversie naar magneetband veronderstellen wij als ingangssituatie de aanwezigheid van ponskaarten of ponsband.

Wederom is de akkurateessekontrolle hierbij van groot belang, mede omdat ook bij technisch zeer betrouwbare apparatuur toch altijd maatregelen moeten worden getroffen voor automatische detektie van verminkingen. Het ligt dan ook voor de hand, mede gezien het te verwaarlozen extra tijdsbeslag op de machine, dat tijdens de konversie naar magneetband kontroletekens en kontrolgetallen opnieuw worden nagerekend. Tot deze kontrolesoort kunnen ook het nagaan van juiste lengte van de ingevoerde bestanddelen en het verifiëren van voorgeschreven blankovelden gerekend worden.

Een andere reeks van controles is die op de volledigheid van de gekonverteerde invoer. Dit kunnen zijn het natellen van het aantal invoerposten, het checken van de aansluiting in volgnummers van de verwerkte posten of partijen en het natellen van kontroltotalen per partij.

Een volgende groep van controlemaatregelen heeft ten doel de geldigheid van de invoer vast te stellen. Dit kan betrekking hebben op de juistheid van de invoermutatie als geheel, zoals een minimaal vereist aantal kaarten of subblokken of een maximaal toelaatbaar aantal kaarten. Een voorbeeld van dit laatste is het toegestaan aantal materiaalregels van een produktspecificatie.

Hierbij moet men attent zijn op het niet deugdelijk werken van deze kontrolle bij verstoring van de volgorde van de invoer. Andere voorbeelden zijn de kontrolle op de geldigheid van de soort ingelezen tekens per specifiek veld en de geldigheid van de inhoud van bepaalde velden. Dit laatste wordt automatisch gecontroleerd wanneer de waarde van het veld kan worden vergeleken met bijv. in het konversieprogramma opgenomen tabellen of met de inhoud van andere invoervelden waarmee een logische samenhang moet bestaan.

9) De term uitschieters wordt in onze werkkring algemeen gebruikt. Het ware wellicht juist te spreken van afkeur, uitval of uitschot, aangezien het uitschieten, d.w.z. het zich onderscheiden van soortgelijke exemplaren hier alleen in ongunstige zin voorkomt. De oorzaak ligt vermoedelijk bij het eenvoudiger spraakgebruik in „vandaag 20 uitschieters” t.o. „vandaag werden 20 mutaties uitgeschoten”.

Een laatste vorm van controle die in dit verband moet worden genoemd is die welke de invoer toetst op plausibiliteit. Men zou ook kunnen spreken van redelijkheids- of waarschijnlijkheidscontrole. Een veel gebruikte engelse term is in dit verband de „limit check”. Voorbeelden hiervan zijn:

- de specificatie (= stuklijst) van een produkt mag niet meer dan bijv. 50 posities bevatten.
- het aantal geregistreerde overuren op een loonkaart mag niet hoger zijn dan 15.
- een incidenteel rabat mag niet meer bedragen dan 25 %.

b. *Uitschieten van partijen*

Bij controle per partij door de computer is het mogelijk een partij die niet konfirmeert aan het kontroletotaal van de partij in zijn geheel te doen uitschieten.

De enige argumenten pro zijn, dat de resterende invoer foutloos wordt en dat het opnieuw invoeren van goede gegevens minder kans op fouten geeft dan het invoeren van korrekties op foutief verwerkte gegevens.

De argumenten kontra wegen o.i. zwaarder:

- de resterende goede invoer wordt tezeer gedecimeerd door sporadische fouten in relatief grote partijen (kleine partijen moeten op hun beurt vermeden worden omdat deze veel extra controle- en konversiewerk vereisen).
- veel, overigens goede, invoer moet opnieuw ingelegd worden.
- bestanden worden te laat bijgewerkt.

c. *Foutenbestand (= error file)*

Om eenmaal ingevoerde gegevens niet verloren te doen gaan en om voortgangskontrolle op korrektieprocedures uit te voeren kunnen uitschieters in een speciaal foutenbestand opgenomen worden. Elke bijboeking moet uiteraard gepaard gaan met het afdrukken van foutsignalen.¹⁰⁾

Een voorbeeld hiervan is invoer, waarbij verminking voorkomt in de inhoud van de signifikante velden, speciaal wanneer deze als sorteerkarakter dienen. Het hierna te bespreken gebruik van fouttekens is in dit geval niet aanbevelenswaard.

De korrektieprocedure omvat een combinatie van het invoeren van een goede mutatie en het afboeken van het foutenbestand.

Periodiek dient een overzicht van nog onafgehandelde fouten afgedrukt te worden.

d. *Machinale correctie*

Ter beperking van het aantal uitschieters kan getracht worden foutieve invoer bruikbaar te maken voor verwerking, uiteraard met behoud van het signaleren van de gekonstateerde afwijking en met vermelding van de automatisch aangebrachte correctie.

Automatisch corrigeren maakt het systeem minder gevoelig voor fouten bij de invoer, terwijl voorts de invoer-procedure sneller kan worden afgewikkeld omdat niet behoeft te worden teruggekoppeld voor het manueel schrijven van korrekties.

¹⁰⁾ In aansluiting op het gewone spraakgebruik wordt de term „signaal” door ons gebruikt in de betekenis van „sein, boodschap, waarschuwingsteken” en dus *niet* in de betekenis die de Informatie- en Communicatietheorie er aan hecht. Wij zien parallellen aan het door ons gebruikte „signaal” met begrippen als uitzonderingsverslaglegging (= exception reporting) en bedrijfs-signalering.

e. Een voorbeeld¹¹⁾ van automatisch corrigeren, dat kan worden bereikt door het gebruik van kontroletekens gekoppeld aan dat van de kontroletelling per partij, hierna te noemen de vertikale natelling, volgt hieronder. In dit voorbeeld onderscheiden wij drie soorten fouten, nl.:

- horizontale of getalfout
- vertikale of kolomfout
- kompenserende fout
- Formule voor kontroletekens: de afzonderlijke cijfers van het nummer, te beginnen bij de eenhedenpositie, vermenigvuldigen met opklimmende machten van 2; de uitkomsten totaliseren, het totaal delen door 11; de rest vormt het kontroleteken, waarbij een rest van 10 wordt weergegeven met het minteken —.
- Formule voor vertikale natelling: sommeer afzonderlijke cijfers per kolom, het controlecijfer is dan gelijk aan de waarde van de eenhedenpositie.

invoer van 4 mutaties	kodenummer	kontrole- teken	berekening
a	1 2 3	0	$\begin{array}{r} 3 \times 2 + 2 \times 4 + 1 \times 8 \\ \hline 11 \end{array} = 2, \text{ rest } 0$
b	2 5 6	4	$\begin{array}{r} 6 \times 2 + 5 \times 4 + 2 \times 8 \\ \hline 11 \end{array} = 4, \text{ rest } 4$
c	6 8 9	-	$\begin{array}{r} 9 \times 2 + 8 \times 4 + 6 \times 8 \\ \hline 11 \end{array} = 8, \text{ rest } 10$
d	1 0 9	4	$\begin{array}{r} 9 \times 2 + 0 \times 4 + 1 \times 8 \\ \hline 11 \end{array} = 2, \text{ rest } 4$
vertikale natelling	0 5 7		$\begin{array}{l} 1 + 2 + 6 + 1 = 10; \text{ dus } 0 \\ 2 + 5 + 8 + 0 = 15; \text{ dus } 5 \\ 3 + 6 + 9 + 9 = 27; \text{ dus } 7 \end{array}$

Indien - slechts - cijfers in één vertikale of horizontale rij fout zijn, kan het computerprogramma trachten de invoermutatie(s) te rekonstrueren.

Horizontale of getalfout

Indien de computer inleest:			zal de computer narekenen:
a	1 2 3	0	1 2 3 0
b	2 5 6	4	2 5 6 4
c	6 8 9	-	6 8 9 -
d	1 9 0	4	1 9 0 0
	0 5 7		0 4 8

De afwijking tussen nagerekend en ingelezen kontroletekens van mutatie d, resp. 0 en 4, wijst op een fout in het kodenummer van deze mutatie.

De vergelijking tussen vertikale kontroletelling '048' en de ingelezen natelling '057' indiceert, dat zowel in eenheden- als tientallenpositie een of meer fouten schuilen:

eenhedenpositie: $3 + 6 + 9 + x = \text{tiental} + 7$; dus $x = 9$

tientallenpositie: $2 + 5 + 8 + y = \text{tiental} + 5$; dus $y = 0$

Vervolgens moet opnieuw het kontroleteken van de gewijzigde mutatie d met kodenummer 109 berekend worden: dit blijkt overeen te komen met het ingelezen kontroleteken 4.

11) Vrij naar Laden & Gildersleeve, „System Design for Computer Applications” blz. 199 - 202.

Vertikale of kolomfout

Indien de computer inleest:			zal de computer narekenen:
a	1 3 3	0	1 3 3 4
b	2 5 6	4	2 5 6 4
c	6 9 9	-	6 9 9 3
d	1 0 9	4	1 0 9 4
natelling	0 5 7		0 7 7

De vergelijking tussen controle- en natelling leert, dat de tientallenkolom fout is; tevens geven de nagerekende controletekens aan, dat de kodenummers in de mutaties a en c fout zijn:

$$\frac{3 \times 2 + x \times 4 + 1 \times 8}{11} = \text{elfvoud} + 0; \text{ dus } x = 2$$

$$\frac{9 \times 2 + y \times 4 + 6 \times 8}{11} = \text{elfvoud} + 10; \text{ dus } y = 8$$

Vervolgens moet opnieuw de controle telling gemaakt worden; deze blijkt nu overeen te komen met de ingelezen natelling 057.

Kompenserende fouten

Het zal de lezer overigens opvallen dat kompenzende fouten uiteraard wel ontdekt, maar helaas niet automatisch gecorrigeerd kunnen worden:

als de computer inleest:			zal de computer narekenen:
a	1 1 3	0	1 1 3 7
b	2 5 6	4	2 5 6 4
c	6 9 9	-	6 9 9 3
d	1 0 9	4	1 0 9 4
natelling	0 5 7		0 5 7

Er wordt gesignaleerd, dat de kodenummers van mutaties a en c fout zijn, maar de natelling geeft géén aanwijzing in welke kolom (of zelfs kolommen) de fout schuilt.

f. Bij de korrektie van de horizontale fout is afgezien van het door het programma laten „proberen” of sprake is van een eenvoudige omzetting van twee cijfers.

Een amerikaanse situatie¹²⁾ leert, dat niet zozeer de zuivere omzettingen van cijfers als wel andere fouten samenhangend met twee cijfers van een getal veelvuldig voorkomen.

Voorbeelden hiervan zijn:

Het getal	werd geponst als
200	20
1000	100
9913	913
12245	11245
3346	3446
2800	28
5870	5800

g. Wanneer door de machine een verschil wordt gekonstateerd tussen de natelling van een controlegetal van één invoer en de ingelezen zgn. horizontale controle telling kan volstaan worden met het schrijven van een signaal onder gelijktijdig aksepteren van de invoer.

¹²⁾ Ontleend aan EDP-Analyzer, vol. 1 no. 1, februari 1963 „New light on error detection and control”.

De oorzaak van de fout kan liggen in het niet goed gewerkt hebben van de controle op de natelling bij het ponsen of bij een vermindering die bij de konversie door de computer is opgetreden. De eigenlijke inhoud van de invoer kan 100 % goed zijn, wanneer de fout schuilt in de ingelezen natelling.

h. Wanneer fouten, d.w.z. ongeldige of ontbrekende tekens, worden ontdekt in bepaalde velden zou eveneens volstaan kunnen worden met signalering hiervan, uiteraard met het afdrukken van de gekonstateerde inhoud van het betreffende veld. Bij het schrijven op magneetband zullen in dit geval speciale *fouttekens* in de betreffende posities van het veld worden geplaatst. Hiermede wordt een speciale behandeling van de invoer ten tijde van het bewerken van het bestand gewaarborgd.

VIII SORTEREN

Het sorteren dient om de invoer in dezelfde volgorde te brengen als die van het bestand. Deze volgorde is een vereiste bij het volgtijdig bewerken van bestand.

De sorteerkennmerken voor de invoer gaan veelal dieper dan die van het bestand. Na het gemeenschappelijk deel kan o.m. gesorteerd worden op mutatiecode en/of op datum.

Op datum om bij wijziging-op-wijziging de jongste invoer als laatste in het bestand te kunnen bewerken; op mutatiecode om basisinvoer zoals het inleggen van een nieuw bestanddeel vooraf te doen gaan aan het verwerken van een mutatie in dit bestanddeel.

Fouten, ontdekt bij één van de geldigheidscontroles worden, indien zij betrekking hebben op ogenschijnlijke onvolledigheid, op simpele wijze automatisch geëlimineerd door bij de sorteerkennmerken ook de volgorde-indikaties op te nemen van de „physical records” die tot één bestanddeel behoren.

IX BIJWERKEN VAN HET AUTOMATIEBESTAND: ALGEMEEN

Mede omdat hierover nog weinig ervaringen gepubliceerd zijn is het van belang te onderzoeken in hoeverre het mogelijk is, onvolmaakte invoer in bestanden te verwerken. Primaire eis moet zijn, dat het bestand voldoende betrouwbaar blijft om de benodigde systeeminformatie te verstrekken.

Aangezien een bestand a.h.w. een levend organisme is waarvan in de praktijk blijkt, dat na verloop van tijd ondanks alle denkbare voorzorgen toch onjuistheden insluipen, is het wellicht beter te streven naar een bestand, dat zo compleet mogelijk de totaliteit van alle invoer weerspiegelt dan naar een bestand dat alleen maar de volmaakte juiste invoer aanvaardt. Een automatische procedure die continu de in het bestand voorkomende onjuistheden opspoot en signaleert is in beide concepties een noodzakelijke component van het systeem.

Het kan ook nog anders geformuleerd worden; De machinesystemen zijn star en beperkt in hun opnamevermogen in vergelijking tot de flexibele, improviserende mens. Door het aanvaarden van niet geheel juiste invoer wordt de volledigheid van de uitvoer-publicatie m.i. dichter benaderd; wordt als het ware het bovengenoemde nadeel ten dele ongedaan gemaakt. Het automatisiesysteem is dan evenals de mens als het ware in staat tot het doen van voorlopige aantekeningen, tot registratie van transacties met een vraagteken.

X BIJWERKEN VAN HET AUTOMATIEBESTAND: DE TECHNIEK

a. Essentieel voor het door middel van computerprogramma's bijwerken van informatiesystemen zijn:

- de indeling van het bestand
- de keuze van de invoersoorten en van de functies, die hieraan worden toegekend.

Hierbij refereren wij aan het gestelde in het begin van dit artikel.

Thans willen wij bij een aantal invoersoorten enige notities plaatsen van de voornaamste te verrichten handelingen. Zoals reeds aangekondigd, gaan wij daarbij uit van de veronderstelling dat niet alleen de invoer zelf maar ook de inhoud van de invoer alsmede de inhoud van het bestand foutief kan zijn. De centrale, hieraan ten grondslag liggende gedachte is, dat voorzover dat machinaal reeds mogelijk is, de ontvangen invoer zo volledig mogelijk geaksepteerd en verwerkt wordt. Dit ontslaat de systeemontwerper uiteraard niet van de dure plicht om tot aan de uiterste grens van het bedrijfseconomisch toelaatbare een stringente *invoer*-kontrole te implementeren; en evenmin mag het gaan betekenen, dat de kwaliteit van de systeem informatie wordt aangetast. Dit laatste houdt o.m. in:

- speciale beveiligingen bij het in het bestand opnemen van niet geheel juiste mutaties;
- duidelijke en volledige (herhalings-)signaleringen van geconstateerde onjuistheden.

b. Ten aanzien van de soorten signalen of meldingen bedienen wij ons van de volgende indeling:

Dynamische of Aktiesignalen, onderverdeeld in:

- werksignalen: signalen die uit de - goede - werking van het systeem voortspuiten en oproepen tot handeling. Deze komen in dit artikel niet aan de orde omdat zij typisch bepaald worden door de doelstellingen van het systeem zelf.
- foutsignalen: signalen bij onvolledige resp. geweigerde behandeling van invoer die het kreëren van korrigerende invoer noodzakelijk maken.

Statische signalen, onderverdeeld in:

- verantwoordingsignalen: signalen die aangeven dat de invoer normaal is verwerkt.
- kontrolesignalen: signalen die aangeven dat de invoer niet of afwijkend is verwerkt zonder dat dit tot korrigerende invoer behoeft te leiden.

Naar de bron kunnen de dynamische signalen nog worden onderverdeeld in signalen die door de invoerverwerking worden opgeroepen en in signalen die uit het bestand voortkomen (bijv. een bepaalde stand of een bepaald tijdstip is bereikt dan wel een fout in het bestand treedt - weer - aan het licht).

c. De invoer kan onvolledig zijn, bijv. de inhoud van één of meer velden ontbreekt of is verminkt. Dit kan al gesignaleerd zijn bij de in paragraaf VII aangegeven invoercontroles onder handhaving van de invoer.

Bij *aanwas* in het bestand is het dan noodzakelijk zgn. fouttekens te plaatsen in de nieuw gekreëerde bestandvelden waarvan de inhoud in de invoer ondeugde-

lijk was onder gelijktijdige herhaling van de reeds bij konversie naar magneetband gegeven foutsignalen. Betreft het invoervelden die de inhoud van een bestandveld hadden moeten *wijzigen* dan dient de invoer geweigerd te worden en met een foutsignaal: „Inkomplete invoer” afgedrukt te worden. Het zal in het algemeen beter zijn in het bestand een verouderd gegeven te handhaven dan dit te vervangen door een foutief gegeven. Wordt daarentegen de invoer gevolgd door een dubbele invoer (zie hierna), dan moet volstaan worden met het afdrukken van de geweigerde oudste invoer tezamen met een controlesignaal.

d. Bij de twee hoofdgroepen invoersoorten „mutaties” en „bestandwijzigingen” is het mogelijk de juistheid van de invoervelden te checken door vaststelling van de gelijkheid van een ander invoerveld-dan-het-sorteerkenmerk en het overeenkomstige permanente veld in het bestanddeel (bijv. een gedeelte van de vaste omschrijving).

Vereist het systeem deze extra controle dan vloeit hieruit voort, dat bij ongelijkheid van deze aanvullende vergelijking de invoer geweigerd moet worden en met het foutsignaal: „Tweede kenmerk wijkt af” afgedrukt wordt.

e. Fouttekens die voor de eerste maal in het bestand worden opgenomen, worden gemeld met behulp van foutsignalen, behorend bij de betreffende invoer.

Na het verwerken van alle invoer inzake een bepaald bestanddeel dient gecontroleerd te worden of het bestand nog fouttekens bevat in andere, niet gewijzigde velden. Zo ja, dan dienen de betreffende velden afgedrukt te worden met het herhalingsfoutsignaal: „Fouttekens in bestand”.

XI BESTANDWIJZIGINGEN

a. Controle op de invoer heeft meerdere aspecten.

Twee algemene aspecten die bij de groep bestandwijzigingen aan de orde komen nl.:

- kwaliteit van de invoer
 - konfrontatie met vorige invoer in dezelfde rekengang
- zullen nu worden behandeld.

Een derde aspect, nl.:

- konfrontatie met bestand,
- komt hierna per invoersoort aan de orde.

Het volgende principeschema geeft een vorm van behandeling van de eerste twee controle-aspecten.

Opgemerkt moet hierbij worden, dat verondersteld is, dat de invoer gesorteerd is op achtereenvolgens:

- sorteerkenmerk
- mutatiecode (= invoersoort)
- creatie-datum van elke afzonderlijke invoer.

„Dubbel” betekent in dit schema: „gelijk sorteerkenmerk, gelijke mutatiecode”.

Bij bestandwijziging moet speciale aandacht aan dubblures besteed worden omdat enerzijds per sorteerkenmerk slechts één bestanddeel in het bestand mag

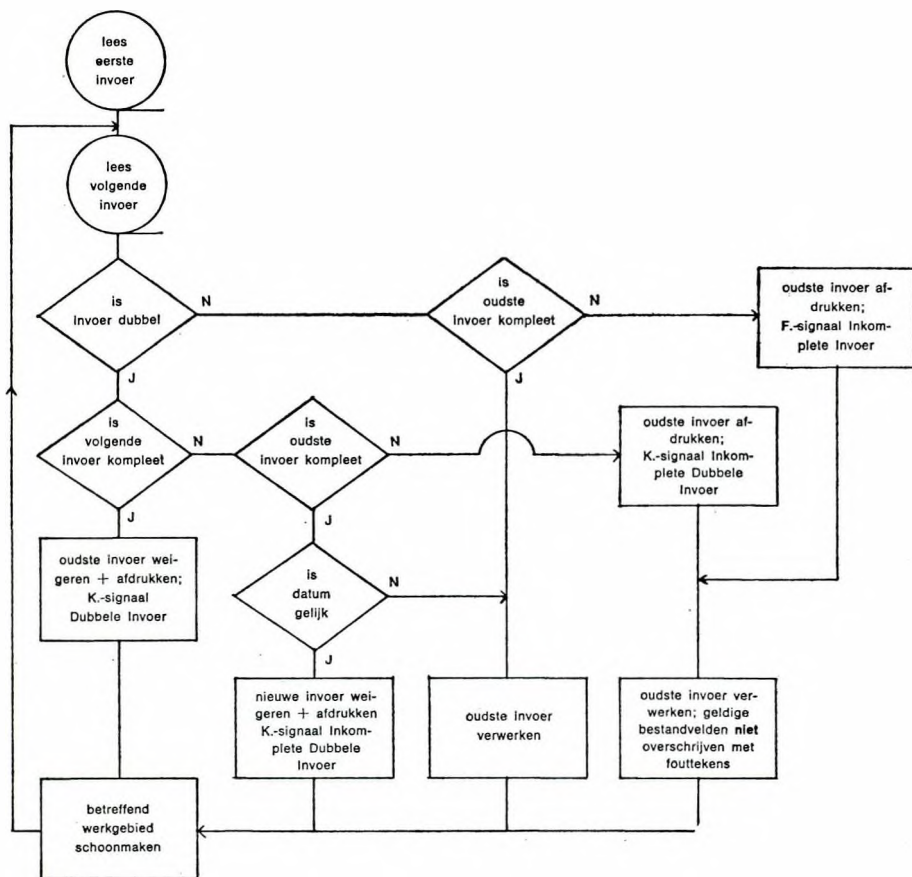
voorkomen en anderzijds omdat bij meerdere wijzigingen van dezelfde velden alleen de meest recente gegevens van belang zijn.

b. Bij dubbele invoer *van gelijke datum* kan de computer niet bepalen welke invoer de juiste is. Indien het programma hiervoor nog ruimte laat, kan worden nagegaan of de invoeren identiek zijn.

Zo ja, dan kan één van de dubbele invoeren worden geweigerd en afgedrukt met een controlesignaal „identieke invoer zelfde datum” dat slechts als kritiek op de invoerorganisatie fungeert. Zijn de invoeren niet gelijk, dan kan het systeem geen logische keuze doen. De mogelijkheden zijn: beide invoeren weigeren of lukraak één van beide opnemen met foutsignaal „dubbele invoer gelijke datum”.

In het volgende schema is deze - gekompliceerde - procedure *niet* gevolgd.

Principeschema bestandwijziging



c. Toelichting op het principeschema.

Er is gebruik gemaakt van twee alternatieve werkgebieden voor het inbrengen van de invoeren. Het kan voorkomen, dat het „tweede” gebied is schoongemaakt terwijl oudere, d.w.z. eerder ingebrachte, invoer in het andere gebied nog onverwerkt is.

Er is weinig verschil tussen de kontrolesignalen „Dubbele Invoer” en „Inkomplete Dubbele Invoer”. Het woord „Inkomplete” geeft alleen een kritiek op de invoerorganisatie. In het geval van „Dubbele Invoer” is niet getest op compleetheid van de invoer en moest de toevoeging dus achterwege blijven.

d. Een voorbeeld van invoer, volgens het prinsipeschema:

In een bestand is een bestanddeel aanwezig met 4 velden, inhoud:	a b c d
1e invoer, met 2 fouttekens	F F e g
2e invoer, dubbel, met 2 fouttekens	h F F j
Situatie in bestand na verwerking:	h b e j

Signalen:

K-sigitaal (= kontrolesigitaal) 1e invoer: „F F e g Inkomplete Dubbele Invoer”

F-sigitaal (= foutsigitaal) 2e invoer: „h F F j Inkomplete Invoer”

Het foutsigitaal leent zich voor opneming in het zgn. Foutenbestand.

XII VOLGORDE VAN INVOERSOORTEN IN EEN REKENGANG

a. Dat de keuze en de definiëring van de funktie van de verschillende invoersoorten van groot belang is kan niet genoeg gesteld worden. Als zenuwstelsel voor de informatieverwerking binnen het systeem dient een sluitend geheel ontworpen te worden. Het zou bijzonder duur in werking en onoverzichtelijk in systeembesturing zijn wanneer men de mutatiecodes a.h.w. spontaan liet ontstaan; men zal vooraf een gefundeerde keuze moeten doen en zich moeten hoeden voor een voor de invoerstations onhanteerbaar groot aantal mogelijkheden. Ook moet men zich bewust zijn dat wijziging qua aantal of qua betekenis van de invoersoorten zeer moeilijk zal zijn.

De volgorde van kodering van de invoersoorten is bepalend voor de sortering en dus ook voor het bijwerken in het bestand. Deze volgorde van verwerking dient grondig door de systeemanalist bestudeerd te zijn omdat deze op zijn beurt weer van invloed is op de toe te kennen funkties van de verschillende invoersoorten.

b. Een voorbeeld ter verduidelijking:

Voor de beide funkties van bestandwijzigen:

- lichten van bestanddeel
- toevoegen van bestanddeel

kunnen twee afzonderlijke mutatiecodes gekreëerd worden.

Het invoeren van deze beide soorten is noodzakelijk indien een bestaand bestanddeel *vervangen* moet worden.

Een andere konsekwentie van deze systematiek is, dat het onmogelijk is een reeds doorgegeven invoer in *dezelfde* rekengang ongedaan te maken met een lichtinvoer. Immers de invoersoort „Lichten” heeft een lagere mutatiecode en dient zich eerder dan de bestandwijziging „Toevoegen” voor verwerking aan.

Deze situatie kan men vermijden door het kreëren van een invoersoort:

- toevoegen of vervangen van bestanddeel.

Indien dan als volgorde van verwerking in de mutatiecodenummers wordt vastgelegd:

- 1 toevoegen of vervangen van bestanddeel
- 2 wijzigen van sorteervelden
- 3 lichten van bestanddeel

zien wij echter een ander mogelijk nadeel voor de interne systeembewaking optreden, namelijk dat een controle op de materiële juistheid van de invoersoort „toevoegen van bestanddeel” wegvalt. Als in een invoer van deze soort een foutief sorteerkennmerk voorkomt dat als zodanig het kenmerk vormt van een reeds bestaand bestanddeel, dan wordt met de invoersoort „toevoegen of vervangen” in dit geval een goed bestanddeel met een daarbijbehorend sorteerkennmerk gelicht en wordt een ander, nieuw bestanddeel aangewassen met een foutief kenmerk.

Aan dit voorbeeld kan ook weer de samenhang tussen de controlebehoeften gedemonstreerd worden: zou bij de invoer, bijv. door controletekens extra zekerheid zijn verkregen ten aanzien van de juistheid van het sorteerkennmerk dan is er weinig kans op de hier geschetste onaangename situatie.

XIII BIJWERKEN VAN BESTAND PER INVOERSOORT

a. Het voert te ver om in het kader van dit artikel een gedetailleerd model uit te werken van de behandeling van de verschillende invoersoorten. Wij volstaan derhalve met de aantekening dat het bij het systeemontwerp zeer nuttig is, om per invoersoort drie aspecten te analyseren:

- de situaties die tijdens het bijwerken kunnen optreden
- de acties of bewerkingen die per situatie moeten geschieden
- de signalering die per situatie en eventueel per actie moet worden gedaan.

Van groot belang is een weloverwogen keuze van de verklarende of kode-teksten die voor de verschillende signalen moeten dienen. Het blijkt vaak dat dit aan de programmeur wordt overgelaten en dat niet voldoende wordt gelet op duidelijkheid ten gerieve van de gebruiker van de systeemuitvoer.

b. *Eén of meer fasen per rekengang*

Het wijzigen van sorteervelden van bestanddelen creëert zgn. gegenereerde invoer die gesorteerd moet worden op de nieuwe kenmerken. Om deze invoer weer automatisch in het bestand te kunnen opnemen is een tweede verwerkingsfase in dezelfde rekengang (in feite zijn het twee rekengangen onmiddellijk na elkaar) nodig. Gebeurt dit niet, dan moet gewacht worden tot de eerstvolgende periodieke verwerkingscyclus.

De keuze wordt belangrijk wanneer men al zo snel mogelijk de gelegenheid wil scheppen, dat mutaties onder het nieuwe sorteerkennmerk verwerkt kunnen worden.

Indien konsekwent met twee verwerkingsfasen per rekengang gewerkt wordt, schept dit de mogelijkheid dat een bestandwijziging zoals „lichten van bestanddeel” die in de eerste fase uitschiet wegens het niet aantreffen van het bestanddeel wordt overgebracht naar de gegenereerde invoer: Het kan immers zijn dat een in dezelfde rekengang van *nieuwe* sorteerkennmerken voorzien bestanddeel gelicht moet worden.

Indien met *niet meer* dan twee fasen per rekengang gewerkt wordt kan zich het probleem voordoen dat een bestanddeel gelicht onder het oude kenmerk en voorzien van een nieuw kenmerk niet in het bestand kan terugkeren omdat in de 2e fase blijkt dat reeds een bestanddeel onder het nieuwe kenmerk aanwezig is.

Voor deze en soortgelijke situaties kan het aanbeveling verdienen te overwegen of opneming van de geweigerde gegenereerde invoer in een zgn. wachtbestand (= suspense file) zinvol is. Deze oplossing versterkt de volledigheidskontrolle en vermijdt het volledig afdrukken en opnieuw ponsen van de gegevens van het betreffende bestanddeel.

c. *Wijzigen van permanente velden*

De layout van de invoer schept in dit geval problemen:

- moet voor elk bestandveld een aparte invoersoort gekodeerd worden?
- mogen de velden alleen groepsgewijs (of bijv. per physical record) gewijzigd worden?
- moet het wijziging of vervanging van de inhoud van een veld zijn?

Vervanging wil hier zeggen: het bestandveld vervangen door het invoerveld.

Wijziging-in-enge-zin wil zeggen t.a.v. de veldinhoud in het bestand iets toevoegen (optellen) of uitsluiten (afrekken).

Vervanging is onder meer noodzakelijk bij vermindering van bestand of bestand-aanwas, bij principiële verandering van vaste normen of systeemdetaïls en ook bij correctie van een lang bestaande foutieve mutatieverwerking.

Bestandwijzigingen zullen veelal volgens dit vervangingsprincipe werken.

Mutaties zullen veelal volgens het principe wijziging-in-enge-zin werken. Een ander woord hiervoor: muteren.

Deze keuze heeft consequenties.

Bij bestanden met een groot aantal verschillende velden per onderdeel zal getracht worden het aantal invoersoorten hanteerbaar, d.w.z. beperkt te houden.

Enerzijds gebeurt dit door permanente velden alleen groepsgewijs te laten vervangen. Probleem hierbij is dat de huidige inhoud van het bestand niet altijd bekend is. Wanneer slechts een enkele veldinhoud vervangen moet worden moet getoedacht worden dat de overige velden van dezelfde groep onaantast blijven.

In principe zijn twee methoden mogelijk:

- afdruk van het bestand maken en op grond daarvan de invoer completeren met de gegevens die onveranderd moeten blijven.
- speciaal teken „Niet wijzigen” in de betreffende invoervelden opnemen: immers het blanco *laten* van de invoervelden zou het blanco *maken* van de overeenkomstige bestandvelden tengevolge hebben.

Anderzijds kan het aantal invoersoorten beperkt worden gehouden door de inhoud van bepaalde permanente velden uitsluitend vervangbaar te maken en de inhoud van andere permanente velden, afgezien van aanwas of lichten van het bestanddeel, alleen muteerbaar te maken.

Dit heeft een nevenvoordeel voor de interne systeembewaking, speciaal wanneer het gegevens betreft die tot een comptabel systeem behoren: Het is dan veiliger de wijzigingsmogelijkheid uitsluitend via een beperkt aantal comptabele *mutaties* te dirigeren.

Bij het vervangen van permanente velden kan het voorkomen dat het betreffende bestanddeel niet in het bestand aanwezig is. Men moet in dat geval kiezen tussen enerzijds een zgn. blinde bestandaanwas en anderzijds het weigeren van de invoer.

Onder het aanwassen van blind bestand verstaan wij het in het bestand opnemen

van een bestanddeel met het in de invoer voorkomende sorteerkennmerk en met de overige gegevens voor zover deze bekend zijn. De niet in de invoer vermelde permanente velden worden voorzien van fouttekens.

Zou daarentegen de voorkeur worden gegeven aan het weigeren van de invoer, dan leent deze zich voor overbrenging naar het reeds genoemde wachtbestand.

De uiteraard met een overeenkomstig foutsignaal gerapporteerde blinde bestandaanwas heeft het voordeel, dat een kettingreactie in het uitschieten vermeden wordt: mutaties inzake hetzelfde bestanddeel kunnen in dezelfde rekengang althans nominaal worden verwerkt.

Het muteren (wijzigen-in-enge-zin) van permanente velden die een foutteken bevatten kan niet geschieden. In zo'n geval vindt in feite *vervanging* plaats; een kontrolesignaal „Veld x bevatte foutteken” moet aanleiding zijn tot een onderzoek naar de eventueel verloren gegane initiële waarde van het veld.

d. *Wijzigen van fakultatieve velden*

Mutaties inzake het wijzigen van *fakultatieve* velden zullen veelal volgens het vervangingsprincipe werken. In een fakultatief veld zal uit de aard der zaak bijv. niet de stand van een langlopend gegeven bijgehouden kunnen worden.

e. *Slotopmerking*

Ook bij het ontwerpen van een model voor de principiële verwerkingsmogelijkheden zowel van bestandwijzigingen als mutaties moet men een keuze doen uit vele alternatieve mogelijkheden. Van groot belang is daarbij het nastreven van onderlinge samenhang tussen de wijze van verwerking en van signalering in overeenkomstige situaties.

Dit kan enerzijds inhouden dat men zeer ver wil gaan met het aanwassen van zgn. blind bestand op grond van de diverse *bestandwijzigingen* die zich kunnen voordoen.

Anderzijds, zoals bijv. bij invoermutaties inzake het wijzigen (d.w.z. vervangen) van fakultatieve velden kan de verwerking geweigerd worden.

Als voorbeelden van groepen van deze velden noemen wij: een inkoopbestelling, een verkooporder of één positie van een produktspecificatie. Weigering volgt, indien noch de oorspronkelijke velden zelf, noch het bestanddeel (in dit voorbeeld het artikel) in het bestand aanwezig zijn.

M.a.w.: Er wordt ook in deze opzet, waarin wij voor redenen van onderzoek zeer ver willen gaan met het simuleren van de menselijke handelwijze, een duidelijke grens gesteld aan het aanwassen van blind bestand.