

**NAMEN NOEMEN**

---

## NAMEN NOEMEN

---

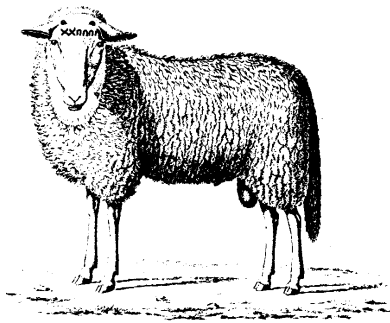
door W. F. Roest

*Dingen die geen naam hebben zijn het noemen niet waard. In de automatiseringswereld wemelt het dus van de namen. Naarmate meer namen genoemd moeten worden, neemt de behoefte aan enige systematiek in de naamgeving toe. Vandaar de talloze standaardnaamgevingsconventies, die allerlei elkaar overlappende deelgebieden dekken. In dit artikel wordt, onder de naam STIC (STandard Identification Conventions) een veelomvattend, maar toch eenvoudig naamgevingsstelsel voorgesteld, waarmee de genoemde overlap kan worden voorkomen. De met STIC toe te kennen namen zijn qua vorm en coderingswijze zodanig geüniformeerd, dat ze ter identificatie van data c.q. system dictionary records kunnen worden gebruikt. In deze records kunnen dan de nodige kenmerken of attributieve gegevens van het benoemde object worden vastgelegd, in plaats van in de naam zelf, via codering. Dergelijke gegevens noemen we metadata. Dit artikel is een bijdrage à titre personnel aan de discussie over standaard naamgevingconventies.*

---

### 1 INLEIDING

Ieder beestje dat een eigen leven leidt, krijgt van zijn baas een naam. Als de belangstelling van de baas beperkt blijft tot het soort beest, dan zal hij volstaan met een soortnaam. Voor de veetransporteur is de soortnaam schaaap veelal voldoende. Wanneer het dier op een bijzondere wijze behandeld, dus geïndividualiseerd, moet worden, dan zal zijn benaming ook unieke identificatie mogelijk dienen te maken. De schaapherder bedenkt dan ook namen als 'Witkopje' en 'Veronica'.



De ene naam is mooier dan de andere; of veelzeggender, of gemakkelijker te onthouden. Het bedenken van mooie, veelzeggende en gemakkelijk te onthouden namen voor unieke objecten vergt onverwacht veel tijd wanneer deze objecten geen markante zintuiglijk waarneembare trekken vertonen. Een naam voor een opmerkelijk uitziend schaaap zal u nu eenmaal eerder te binnen schieten dan een naam voor een abstract iets zoals een stukje informatie.

In de automatiseringswereld komen we dan ook geregeld namen tegen die een vernuftige afkorting vormen van een of ander inhoudelijk aspect van het betrokken object of objecttype. Een lijst met enkele tientallen van dergelijke namen is nog wel te hanteren. Naarmate echter het aantal namen toeneemt, zal de behoefte aan systematiek in de naamgeving toenemen.

Dat kunt U ook waarnemen in de paardenwereld en in de astronomie. In de automatiseringswereld zijn er inmiddels zoveel objecten benoemd en nog te benoemen, dat enige bezinning op naamgevingssystemen de moeite waard is.

## 2 NAMENCHAOS

De verscheidenheid en de veelvoudigheid van te benoemen objecten is zo groot, dat een poging om tot standaardisatie van hun naamgeving te komen welhaast gedoemd is om te mislukken.

We worden geconfronteerd met applicaties, lijnen, nodes, terminals, clusters, remote modems, polling addresses, locaties, domains, projecten, data sets, files, authorities, accounts, DBD's, segmenten, velden, transacties, versies, modules, views, profiles, diskettes, queue's, libraries, users, menu's, exits, index 1 clusters, SSA's, clists, mapsets, edit mark tables, application control blocks.

Deze opsomming is slechts een uiterst summiere bloemlezing uit glossaries en registers van vele manuals, waarvan er heel wat een paragraaf over naming conventions bevatten. We komen, bijvoorbeeld over de naamgeving van een programma, de volgende mogelijkheden tegen.

- \* RxxAV..
- \* P(psbname)
- \* Psvmmmmm
- \* ssftnnn
- \* Xppnnn

Iedere letter heeft daarbij een bijzondere variabele betekenis die allerlei, ongetwijfeld zinvolle, kenmerken weergeeft. Met behulp van de bijbehorende handleiding kunnen we wat meer te weten komen over de plaats, functie, omgeving e.d. van het desbetreffende programma.

De verschillen in de bovenstaande programmanamen komen niet voort uit verschillende karakteristieken van het programma, maar uit de verschillende invalshoeken waarmee een en hetzelfde programma wordt aangeduid. Die invalshoeken zijn bijvoorbeeld die van de programmeertaal, van het database managementsysteem, van het teleprocessing netwerk, van het besturingssysteem, of van de gebruiker.

Een vergelijkbare spraakverwarrende situatie doet zich voor ten aanzien van terminals. Wie hebben te maken met de aanduiding van een bepaalde terminal? En is die aanduiding steeds hetzelfde? We ontmoeten dan, behalve de bekende eindgebruiker, ook de netwerkbeheerder,

de technische onderhoudsdienst, de boekhouder (voor de doorbelasting), en ook VTAM, CICS en het operating system, met allemaal verschillende behoeften aan kenmerkende gegevens over het betrokken terminal.

Een ander voorbeeld van ongemakkelijke naamgeving is te vinden in de uiteenlopende identificaties waarmee een gebruiker toegang krijgt tot verschillende systemen in zijn organisatie. Een gebruiker die verschillende identificaties moet onthouden voor verschillende soorten werk op verschillende systemen, zal zijn geheime passwords overigens zo eenvoudig en eensluidend houden, dat ze in feite vrij gemakkelijk te kraken zijn.

Naast de verschillen in naamgeving voor een het hetzelfde object (zoals voor de bovengenoemde gebruiker) is het opvallend hoe verschillend ook de naamgeving voor verschillende objecttypen is. De formaten en de coderingswijzen van de namen voor bijvoorbeeld een transactie, een veld, of een schijf, vertonen veelal een grote verscheidenheid.

Dit gebrek aan systematiek is een complicerende factor bij het automatiseren van de automatisering. De auteur van dit artikel heeft de afgelopen jaren herhaaldelijk pogingen ondernomen om wat orde te helpen scheppen in de namenchaos, soms met enig, maar meestal met weinig succes.

In het begin spitste de discussie zich toe op de vraag of afgekorte omschrijvende namen (acroniemen of mnemonische namen) op den duur inderdaad meer problemen zouden opleveren dan oplossen.

Die discussie lijkt nu grotendeels achter de rug: gecodeerde namen krijgen nu gewoonlijk de voorkeur. Men is het er ook wel over eens dat de hierboven aangeduide problematiek van per objecttype verschillende coderingswijzen en van verschillende coderingswijzen voor een en hetzelfde object in de praktijk voorkomen zou moeten worden.

Desondanks zijn we nog niet veel opgeschoten. Het ziet er zelfs naar uit dat de namenchaos steeds groter wordt nu steeds meer zaken in geautomatiseerde systemen worden ondergebracht.

Ook nieuwe generaties automatiseerders en gebruikers (doe-het-zelvers) zullen op een zeker moment ontdekken dat al die documenten, tekeningen, modellen, en bestanden die zij zelf in personal computers, CAD/CAM systemen, expertsystemen en office systems hebben ingebracht, zo langzamerhand een onoverzichtelijke massa gaan vormen.

Het veelgebruikte voorbeeld van een jaarverslag leert ons dat voor de benodigde tekst, afbeeldingen en cijfermateriaal uit verschillende bronnen geput wordt, waarbij een behoorlijk opgezette metadata-administratie goede diensten zou kunnen bewijzen.

Dat brengt ons tot de kern van dit artikel: gegevens over objecten in de automatiseringswereld kunnen beter in een metasysteem worden ondergebracht dan in de naam van die objecten. Een naam dient zo goed als niets meer te zeggen over het benoemde.

In de praktijk stuit het invoeren van een nieuw naamgevingstelsel doorgaans op grote weerstand. Wanneer iemand een ding een naam heeft gegeven, verschaft die naam hem/haar een greep op dat ding die ongaarne wordt losgelaten, ook al is een nieuw naamgevingstelsel aanzienlijk fraaier of handiger.

Er wordt ook gewezen op de moeilijkheid dat er al namen zijn die niet zomaar kunnen worden veranderd. Verder is de inspanning die is geïnvesteerd in het bereiken van een zekere consensus over de bestaande standaards, vaak zo groot geweest, dat het onhaalbaar wordt geacht om weer opnieuw te beginnen.

Database administrators, werkvoorbereiders, systeemprogrammeurs, systeemontwerpers en gebruikers zullen het zelden eens worden over een allesomvattend naamgevingsstelsel.

Daar komt nog bij dat diverse softwaresystemen zo hun eigen eisen stellen. We zullen daarom de conclusie moeten trekken dat het omnoemen van bestaande namen vermeden dient te worden.

De in de praktijk bestaande officiële namen en ook de vaak gebezigde roepnamen en bijnamen van bestanden, toepassingen, apparaatuurcomponenten e.d., kunnen niet worden uitgebannen, en hoeven dat ook niet, indien men beschikt over een geautomatiseerd metadatasysteem, waarin de voorkomende synoniemen worden bijgehouden.

Wellicht kan dan ook een befaamde strijdvraag uit de wereld worden geholpen. De voorstanders van, laten we zeggen: 'Algemeen Beschaafd Semantisch' (omschrijvende namen van soms wel meer dan twintig tekens; maar te omslachtig bij het formuleren van interactieve queries), worden niet belemmerd in hun voorkeur.

Evenmin als de liefhebbers van 'Ongeordend Kort Pragmatisch' (geheugensteuntjes die gemakkelijk uitspreekbaar en intikbaar zijn, maar op den duur niet uniek, en zonder vast formaat).

De in het navolgende behandelde STandard Identification Conventions (STIC) kan dienen als een soort 'Eenvoudig Standaard Lexicaal', waarmee verschillende in zwang zijnde of komende namen eenduidig kunnen worden thuisgebracht via een uniek dictionary record. Afhankelijk van de omgeving (een batch programma, een interactieve query of een gesprek met een gebruiker), kan dan de meest geschikte naam, roepnaam of bijnaam worden gebruikt.

### 3 STIC

Met het toenemen van het aantal reeds bestaande namen vergt het kiezen van een nieuwe unieke naam meer en meer tijd. Praktische snuffjes ter herkenning van unieke beestjes in het automatiseringsdomein vormen op den duur een wonderlijke lappendeken waarvan het onderhoud steeds moeilijker wordt en waarbij de kans op fouten toeneemt.

Indien we een in beginsel allesomvattend maar toch eenvoudig naamgevingstelsel willen hanteren, dan ontkomen we er niet aan onze wensen ten aanzien van de inhoudelijke betekenis te laten vallen.

Een naam als VERK-PNT-ADR-MUTCOD is trouwens niet alleen moeilijk uit te spreken, maar ook lastig in te tikken.

Naast te stellen eisen van uitspreekbaarheid en intikbaarheid dienen we te letten op de kans op fouten als gevolg van verwisseling van i en l en O en 0. De automatiseerbaarheid wordt gediend met een eenduidig formaat, waarbij de naam bij voorkeur begint met een letter, geen speciale tekens bevat, en korter is dan negen posities.

Teneinde te beschikken over voldoende benoemingsruimte zal een naam echter al gauw tenminste zes tekens lang dienen te zijn. De meest eenvoudige en afdoende oplossing ligt dan voor de hand:

**XNNNNN** of **XNNNNNN** of **XNNNNNNN**

Dat betekent een willekeurige letter, gevolgd door vijf, zes of zeven volgnummers, afhankelijk van de omvang van het bedrijf of van de instelling waarvoor het naamgevingstelsel is bedoeld.

Deze namen kunnen gemakkelijk dienen als identificatie van system dictionary records waarmee alle nuttige attributieve gegevens kunnen worden vastgelegd, bijgehouden en weergegeven. De hierboven genoemde identificatiewijze is volledig betekenisloos.

Zo valt het uit E69771 niet op te maken of hiermee een gebruiker, een terminal of een programma wordt geïdentificeerd. Misschien zullen we in de toekomst inderdaad deze uiterste consequentie trekken uit de tot dusverre gevolgde redenering.

STIC echter gaat niet zover. Weliswaar is in de loop der jaren steeds meer betekenis verdwenen uit via STIC verkregen namen (STICCERS, spreek uit: stickers), maar enige resten zijn toch achtergebleven.\*

Een STICCER is de identiteit van een entiteit in het automatiseringsdomein, waarbij onder 'entiteit' wordt verstaan: een telbaar object van beschrijving met een eigen levensloop (bijvoorbeeld in een system dictionary).

In STIC worden deze objecten van beschrijving ingedeeld in veertien hoofdklassen, waarvan er acht zijn onderverdeeld in subklassen (zie de schema's 1 t/m 9). STICCERS bestaan uit zeven tekens: twee classificerende letters en vijf identificerende cijfers:

**XXNNNNN**

In diverse gevallen is dit formaat niet overeenkomstig de voorschriften van bepaalde softwaresystemen. De namen van CICS-

\* vgl. eerdere versies van STIC gepubliceerd in:  
- COMGE-bulletin nr 9, 1980  
- GR14-5912, 1982, IBM Nederland N.V.

transacties en van STAIRS-databases mogen niet langer zijn dan vier tekens. Tapes en disks dienen meestal met zes posities te worden geïdentificeerd.

In de praktijk zal voor die gevallen een keuze moeten worden gemaakt uit diverse mogelijke kunstgrepen. Onder handhaving van de volledige identificatie voor het system dictionary record, kan bijvoorbeeld een deel van de STICCER worden weggelaten voor de identificatie van het object voor het betrokken softwaresysteem.

Een bespreking van de in STIC toegepaste of mogelijke classificatiecriteria, en van de geclassificeerde objecten zelf, blijft hier achterwege. We zouden daarbij op teveel automatiseringstechnische details moeten ingaan.

Er kunnen overigens redenen bestaan of ontstaan om de hier in schema gebrachte indeling te herzien. Ook de gebruikte benamingen kunnen desgewenst worden veranderd. De nog niet ingedeelde hoofdklassen kunnen nog volgens toepasselijke criteria worden onderverdeeld.

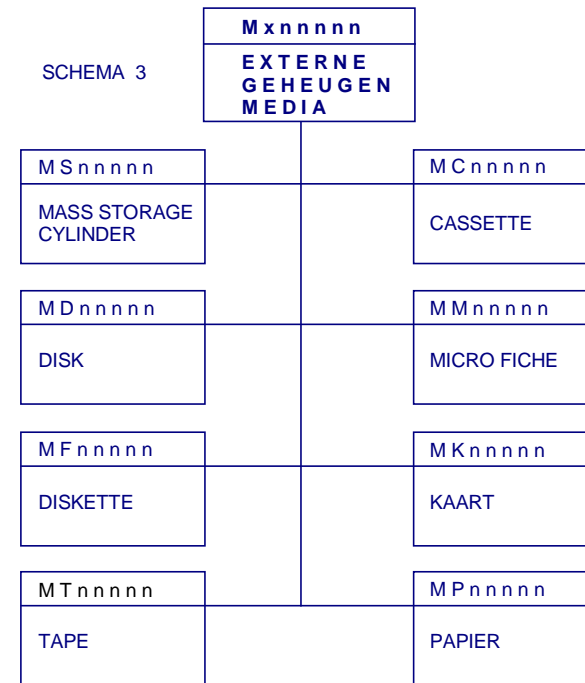
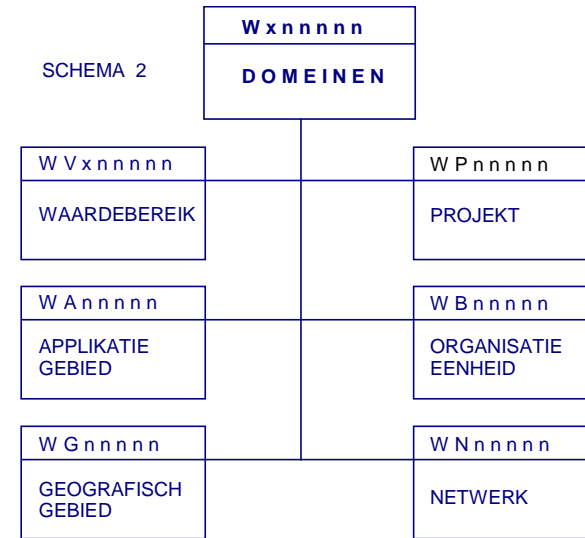
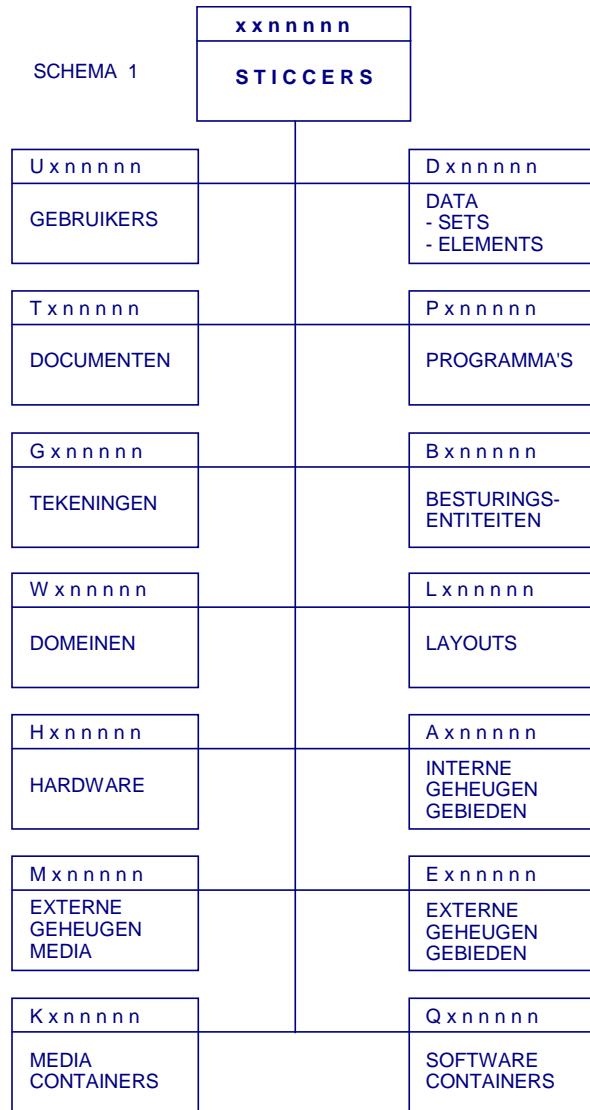
Soms is het nuttig of noodzakelijk om aan te geven of het geïdentificeerde object

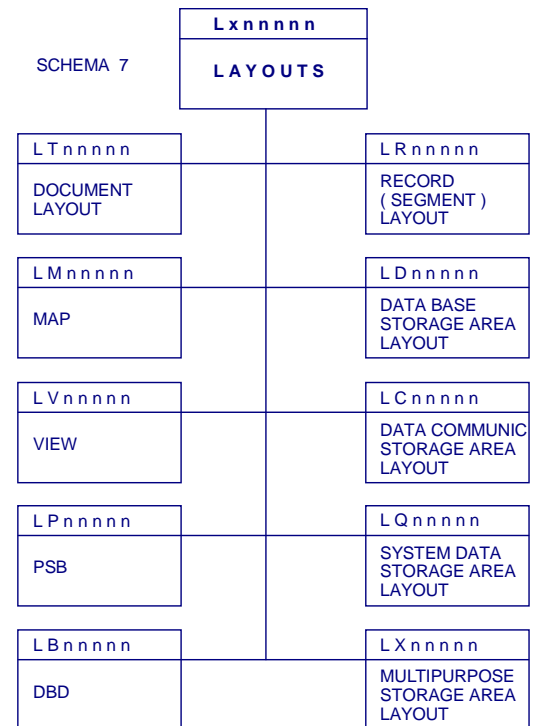
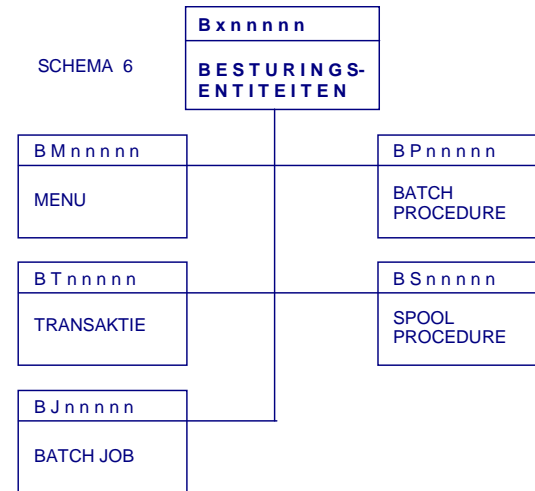
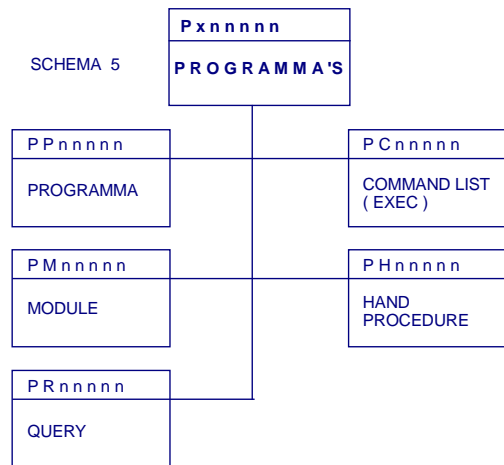
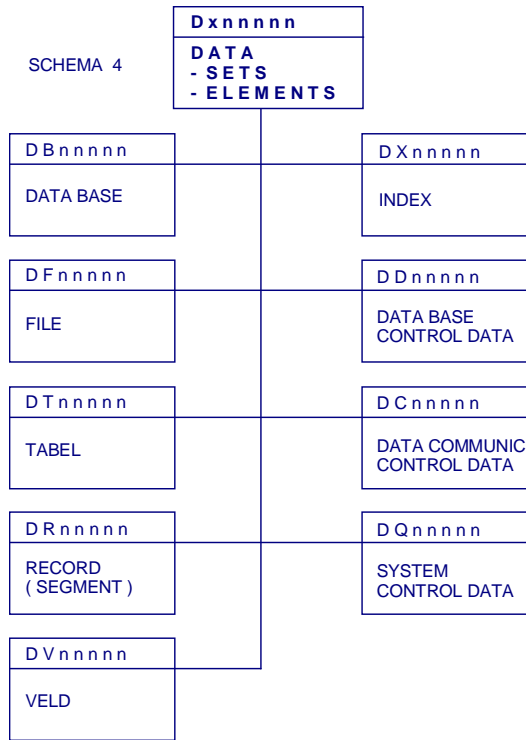
- de 1e, 2e, 3e enz. versie is
- invoer of uitvoer betreft
- een test-, productie-, of herstelvariant aanduidt.

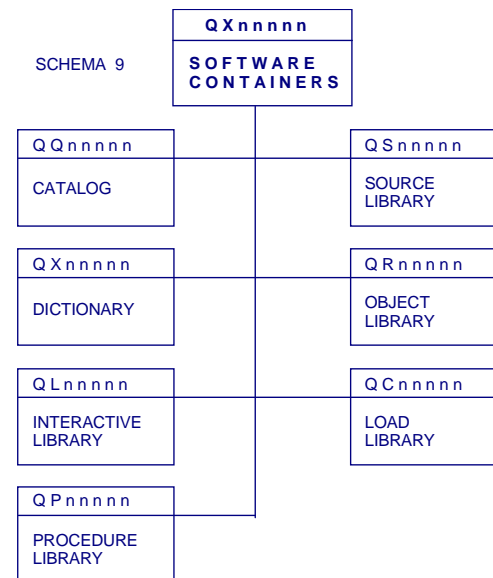
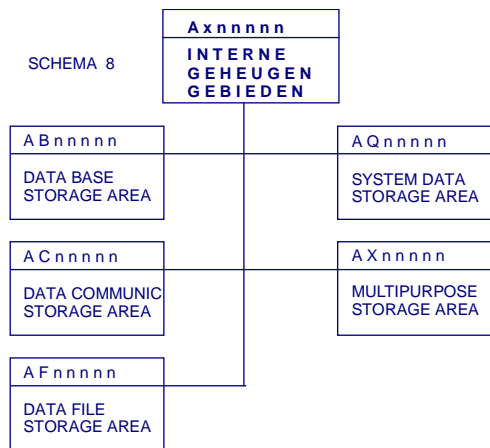
De STICCER kan daartoe worden uitgebreid met een vaste of tijdelijke volgletter.

Verder kan de database- of system administrator de eerste duizend nummers voor zichzelf reserveren. In feite gaat men dan de betekenisloze nummers toch weer een inhoudelijke betekenis geven, hetgeen naar valt aan te nemen, op den duur niet is vol te houden.

In sommige gevallen zijn STICCERS overbodig. Wanneer de naam van een object niet verder bekend hoeft te zijn, dan is een omschrijvende naam, of soms zelfs een willekeurige afkorting afdoende. Dat kan gelden voor, bij voorbeeld: een lokale variabele binnen een module, of voor een weggooprogramma.







#### 4 METADATA

Metadata zijn gegevens over gegevens. Meestal bedoelen we ook gegevens over programma's. En ook: gegevens over de bevoegdheid om programma's te gebruiken die metagegevens kunnen veranderen.

Eenvoudigheidshalve wordt hier volstaan met het noemen van enkele mogelijk zinvolle gegevens voor een system dictionary record:

- roepnaam
- toepassingsgebied
- versie
- installatiedatum
- auteur
- samenstellende onderdelen
- gebruikers
- kwantitatieve gegevens
- relaties met andere entiteiten
- bijnamen
- omschrijving
- data laatste versies
- status(sen)
- taal, layout, waardebereik
- beveiligingsgegevens
- aanschafwaarde, boekwaarde
- invoer- en uitvoerverbanden

Relaties tussen de onderscheiden objecten van beschrijving kunnen in het system dictionary record worden opgenomen in de vorm van de STICCERs van de gerefereerde entiteit.

Wanneer er veel relaties beschreven moeten worden, (bij voorbeeld tussen velden en records of tussen werkmaatschappij en programma's) dan kan het zinvol zijn om deze relaties onder te brengen in een aparte tabel, en daar in zijn geheel naar te verwijzen.

Het beheer van al deze via STIC geïdentificeerde metagegevens kan worden geautomatiseerd met behulp van DB/ DC Data Dictionary (5740-XXF of 5746-XXC), SQL/DS (5748-XXJ), DB2 (5740-XYR), of met een ander in aanmerking komend interactief invoer-, mutatie- en retrievalsysteem.

#### 5 SLOT

Door de vorm van standaards te vereenvoudigen en hun onderlinge overlapping te elimineren, wordt het automatiseren van de automatisering vergemakkelijkt.

STIC, of een naar goedgevonden te bepalen variant ervan, kan daar wellicht een bijdrage toe leveren.

De auteur is de heren A. M. van Rooijen en G. Roza erkentelijk voor hun waardevolle opmerkingen.