

VG-PACK
Programmers
Manual

J.T. van der Wal

Inhoudsopgave

1.	<u>Oorsprong VG-Pack</u>	1
1.1.	<u>Aanleiding tot VG-Pack.</u>	1
1.2.	<u>Doelstelling:</u>	1
2.	<u>Gebruikte Gereedschappen</u>	1
2.1.	<u>De computer:</u>	1
2.2.	<u>De programmeertaal:</u>	1
2.2.1.	<u>De programma's:</u>	2
3.	<u>Programmeerervaringen met VAX/VMS Pascal.</u>	2
3.1.	<u>Programmeren van de VAX8600.</u>	2
3.2.	<u>Enige aanwijzingen en tips voor programmeurs.</u>	4
3.2.2.	<u>Inlezen van strings.</u>	5
3.2.3.	<u>Inlezen van gegevens vanaf files.</u>	5
3.2.4.	<u>Keuzebepaling op basis van strings.</u>	6
3.2.5.	<u>Filenamen</u>	6
3.2.6.	<u>Omgaan met en dimensionering van arrays.</u>	7
3.3.	<u>Suggesties ter verbetering van VG-Pack.</u>	8
3.4.	<u>Van Turbo naar VAX-Pascal.</u>	9
4.	<u>Omgaan met de VAX8600 als computersysteem.</u>	10
4.1.	<u>Handig gebruik van de VAX.</u>	10
4.1.1.	<u>Conditionele uitvoering van één of meerdere opdrachten.</u>	11
4.1.2.	<u>Interactieve aanroep van programma's vanuit com-files.</u>	12
4.1.3.	<u>Gegevensbescherming.</u>	12
4.1.4.	<u>Terminal</u>	13
4.1.5.	<u>Hulpfaciliteiten.</u>	14
4.2.	<u>Omgaan met de Vax8600.</u>	14
4.2.1.	<u>Overzicht Bestanden</u>	15
4.2.2.	<u>Copiëren en verwijderen van Bestanden</u>	15
4.2.3.	<u>Wijzigen van bestanden</u>	16
4.2.4.	<u>Printen van bestanden</u>	16
4.2.5.	<u>Het Password</u>	17
4.3.	<u>Foutmeldingen op de VAX</u>	17
5.	<u>Literatuurlijst.</u>	20
	Lijst van Figuren.	22
	Lijst van Tabellen.	23
	<u>Index</u>	24

1. Oorsprong VG-Pack

1.1. Aanleiding tot VG-Pack.

Op het Biologisch Centrum van de Rijksuniversiteit van Groningen, is een aantal computerprogramma's beschikbaar bij de Vakgroep PlantenOecologie voor het verwerken van vegetatiekundige datasets. Een aantal van deze programma's is ontwikkeld door L. Fresco, wetenschapper bij de Vakgroep PlantenOecologie. Andere programma's zijn afkomstig van o.a. het IWIS-TNO te Wageningen en de Cornell University, Ithaca, New York.

Door het ontbreken van hulpfaciliteiten en de afwezigheid van overzicht van de verbanden tussen de verschillende programma's was het systeem moeilijk toegankelijk. Iemand die van de programma's gebruikt maakte was voor hulp afhankelijk van anderen met meer ervaring. Door het ordenen van de programma's in een menu-gestuurd systeem, waarbinnen enige hulp geboden wordt, is het mogelijk om de mate van gebruik van de programma's te vergroten.

1.2. Doelstelling:

Programmeren van een bruikbaar menugestuurd systeem voor computerondersteunde verwerking van vegetatiekundige datasets.

2. Gebruikte Gereedschappen

2.1. De computer:

Oorspronkelijk waren de programma's voor deze gegevensverwerking geprogrammeerd voor de Cyber 170. Deze computer zal volgens de planning omstreeks Kerst 1987 vervangen worden. En volgens het RekenCentrum van de Universiteit (in januari 1987) is het heel goed mogelijk dat de opvolger(s) geheel anders zullen zijn. Een uitgebreide programmeerinspanning op de Cyber zou dus voor slechts een jaar nut hebben. Daarom is ervoor gekozen om het geheel (althans een zo groot mogelijk deel) tijdens dit onderwerp op de VAX8600 van het RekenCentrum te implementeren. Deze computer is in 1986 in gebruik genomen en maakt deel uit van een uitgebreide reeks minicomputers. Hetgeen ook voor de verdere toekomst enige zekerheid biedt dat het gedane werk verder ontwikkeld kan worden zonder volledige herprogrammering noodzakelijk te maken.

2.2. De programmeertaal:

Voor het meerendeel zijn de programma's waar het om gaat in Pascal geschreven. Enkele programma's zijn in Fortran geschreven. Voor het te programmeren menusysteem is van Pascal gebruik gemaakt.

De oorspronkelijk versies van de programma's (zoals ze op de Cyber 170 staan) waren in het Pascal voor die machine geschreven. Voor de VAX8600 is een versie van Pascal gebruikt, die de mogelijkheden van de VAX volledig kan benutten. Dit Pascal heeft meer standaardmogelijkheden dan het Pascal van de Cyber. Veel van deze extra's zijn in de loop der tijd ook voor de Cyber gepro-

grammeerd, en een programma dat hetzelfde doet is ook zeker mogelijk voor de Cyber 170.

2.2.1. De programma's:

Tijdens onderwerp zijn een aantal nieuwe Personal Computers beschikbaar gekomen op het Lab.voor PlantenOecologie. De meeste programma's die in het menugestuurde systeem zijn opgenomen zijn van de P.C. afkomstig. In feite zijn de programma's op de VAX8600 allemaal afkomstig van een vroege versie van VEGROW. VEGROW is de P.C.-tegenhanger van VG-PACK en is in Turbo-Pascal geprogrammeerd. Bij het maken van VEGROW zijn de oorspronkelijke Cyber 170 programma's als startmateriaal gebruikt.

De snelste en meest efficiënte methode om deze programma's op de VAX8600 weer aan de gang te krijgen is: de belangrijkste verschillen tussen VAX- en Turbo-Pascal op te vangen door het toevoegen van enkele procedures.

De verschillen tussen de P.C. en de VAX zijn veelal opgevangen door op de VAX enkele procedures toe te voegen aan de programma's, om de verschillen adequaat ondervangen.

Deze toegevoegde procedures zijn als bijlages bijgevoegd.

Getracht is de verschillen tussen de programma's voor de P.C. en de VAX minimaal te houden. Om een wisseling van systeem niet onnodig ingewikkeld te maken. Het cruciale verschil tussen de P.C. en de VAX is dan de grote van de datasets die verwerkt kunnen worden.

De oorzaak hiervoor is de ruimte die beschikbaar is tussen het grootste negatieve natuurlijke getal en de positieve tegenhanger daarvan (deze getallen worden vaak aangeduid als resp. "-MAXINT" en "MAXINT"). Waardoor grotere arrays beschreven kunnen worden, die in het virtuele geheugen verwerkt kunnen worden.

Tabel 1: Verschil in maximale integer tussen P.C. en VAX.

	P.C.	VAX
MAXINT	32768 (2^{15})	2147483647 (2^{31})

Hierbij is het aantal bits dat de computer gebruikt om mee te rekenen, ookwel de woordgrootte, bepalend. Voor de P.C. is dat 16 bits (15 voor het getal + 1 voor het teken), de VAX gebruikt hiervoor 32 bits (31 + 1).

3. Programmeerervaringen met VAX/VMS Pascal.

3.1. Programmeren van de VAX8600.

De VAX is, een handigere computer om te moeten programmeren dan de Cyber. Een P.C. programmeren met Turbo-Pascal is nog gemakkelijker. Het voordeel van de VAX t.o.v. de Cyber is dat deze

computer uit geschikter is voor interactief gebruik. De uiterlijke verwantschap van het bedrijfssysteem van de VAX - VMS- , met MS/PC-DOS maakt de omgang met systeem voor mensen die al enige ervaring hebben met P.C.'s makkelijker.

Belangrijker, in het kader van dit onderwerp, is dat de VAX door het virtuele geheugen veel grotere arrays e.d. kan verwerken dan de Cyber.

De VAX is ook geschikt voor BATCH-toepassingen. En ook interactieve programma's kunnen vanuit Batch gebruikt worden; alle antwoorden moeten dan in het aanroepende file aanwezig zijn.

Bij het oplossen van problemen met Pascal programma's is Maarten van Gelderen, werkzaam bij het RekenCentrum, een goede vraagbaak gebleken. Van zijn hulp heb ik goed gebruik kunnen maken.

Ook heeft Maarten van Gelderen een library met nuttige c.q. handige procedures en functies. (In directory [gurv.maarten]¹, de betrokken files heten pasproc.XXX (XXX is een "extensie", o.a. HDR en OLB))

Enkele van deze procedures zijn in de programma's gebruikt n.l.: NextCommand, een procedure die bij het verlaten van een Pascal-programma vanuit dat programma een opdracht in VAX/VMS meeneemt en deze door de VAX laat uitvoeren (b.v. een opdracht om een ander programma te starten.);

WaitForReturn, een procedure die de boodschap "Press RETURN to continue" op het scherm geeft en wacht met verdere uitvoer totdat er op RETURN is gedrukt.

ClrScr, is een procedure, die equivalent is aan de ClrScr-opdracht van Turbo-Pascal, het scherm wordt dus gewist. Om goed te kunnen werken moet deze procedure echter wel kunnen nagaan met wat voor type terminal gewerkt wordt. Dit wordt gedaan door te kijken wat de waarde is van een 'symbol' termtyp. Is dit symbol niet gedeclareerd dan wordt het programma afgebroken en volgt een foutmelding. De opdracht om dit symbol te declareren is:

termtyp:=tvi912 of b.v. vt100, of tvi925, afhankelijk van de terminal die gebruikt wordt.

PASPROC.hdr bevat alle headers van de procedures en functions, en een external-declaration. Aan de hand van dit file kan gemakkelijk gekeken worden of er een oplossing is, en hoe die oplossing dan moet worden aangeroepen. Een andere mogelijkheid om dit file te gebruiken is de tekst ervan op te nemen in de programma zodat alle PASPROC-routines beschikbaar zijn.

PASPROC.olb is het file waarin, in library-vorm, alle routines in gecompileerde vorm zijn opgenomen. Tijdens het linken moet dit file ook opgegevens worden wanneer PASPROC-routines gebruikt worden.

¹ Het directory [gurv.maarten] bevindt zich op de eerste drive van de VAX8600. Om vanaf een andere disk-drive dit directory te kunnen bereiken moet ook de drive worden aangegeven. Hiervoor moet voor het directory de volgende diskomschrijving worden opgenomen: 'disk\$1:'. Het filegebied van PlantenOecologi bevindt zich op Disk 4.

Een Pascal compile opdracht is als volgt:

pascal programma

Resultaat van deze opdracht is dat de broncode van het programma uit het file "programma.pas" wordt omgezet in een file met "object-code", dit file heet dan "programma.obj".

Daarna moet het gecompileerde programma gelinkt worden:

link programma + disk\$1:[gurv.maarten]pasproc/lib

Van de file "programma.obj" wordt een gelijknamig file gemaakt met een .exe-extensie. Aan dit exe-file worden de gebruikte routines uit het library-file toegevoegd en ook nog de zogeheten "Run-Time Library". Hetgeen aan het eind van deze bewerking in het file "programma.exe" staat is een programma dat met de opdracht run programma tot uitvoering kan worden gebracht.

De consultatiedienst van het Rekencentrum heeft mij ook geholpen met het oplossen van enkele problemen. Als ze het daar niet zelf weten kunnen ze je doorverwijzen naar de juiste persoon voor het oplossen van het probleem.

3.2. Enige aanwijzingen en tips voor programmeurs.

Naar aanleiding van problemen, die tijdens het werk optraden, maar die ondertussen zijn opgelost, worden enige aanwijzingen en tips gegeven. Hierdoor kan bij verdere vervolmaking/uitbreiding van het pakket optreden van dezelfde problemen voorkomen worden. Het voordeel van deze beschrijving is dat andere programmeurs die op de VAX met Pascal gaan programmeren, niet weer dezelfde problemen hoeven op te lossen. Hierdoor kan verbetering van VG-Pack ook sneller gebeuren.

Voor zover mogelijk zijn alle fouten en problemen opgelost. Problemen die niet opgelost konden worden zijn zodanig omzeild dat geen fouten meer optreden. Dit is geenzins een garantie voor een vlekkeloze, foutvrije werking van de programma's. Veel fouten komen pas in de praktijk aan het licht.

Voor alle bekende problemen is tot nu toe een oplossing gevonden, of is in ieder geval een methode om het probleem te omzeilen.

Van elk onderwerp dat tot een probleem aanleiding gaf, wordt omschreven: wat het probleem is,

de reden waardoor het ontstaat -indien bekend-
en hoe het probleem kan worden opgelost, dan wel hoe het probleem kan worden omzeild.

3.2.1. Strings op de VAX.

Strings moeten op de VAX door de programmeur als type gedeclareerd worden. En wel als volgt:

(voorbeeld)

```
TYPE string = VARYING [80] of CHAR;
```

```
VAR regel: string;
```

De VAX kent geen string-type als Turbo-Pascal, en packed arrays of character voldoen ook niet.

Varying of char is geschikt om een string mee te declareren. Daarbij moet een lengte worden opgegeven. Rechtstreeks een variabele als varying of char declareren is vanzelfsprekend ook mogelijk.

```
VAR      FILENAME: VARYING [40] of CHAR;
```

Het is mogelijk om na te gaan wat voor teken op een bepaalde plaats in een string staat:

```
If Filename[6]=':' then .....
```

3.2.2. Inlezen van strings.

Bij het lezen van strings ging de VAX vaak door zonder te stoppen en de gebruiker de kans te gunnen om een antwoord op een gestelde vraag te geven.

Het inlezen van strings kan het beste gedaan worden m.b.v. het READLN-statement. Ook voor andere leesopdrachten van INPUT heeft dit de voorkeur.

De reden hiervoor is dat de VAX leest tot het eerste teken dat niet meer past bij de in te lezen variabele. Daardoor laat de computer de Return of EOLN-code nog in het inputbuffer staan. Bij een volgende leesopdracht staat dat er nog en is de VAX direct klaar met de leesopdracht. Bij gebruik van READLN wordt dit probleem vermeden, doordat dan ook de EOLN-code wordt weggelezen tijdens de leesopdracht. En dus verdwenen is wanneer nog een leesopdracht volgt.

3.2.3. Inlezen van gegevens vanaf files.

Het inlezen van gegevens vanuit een file leidde in eerste instantie vaak tot een foutmelding door te lang doorlezen van een file. De oorzaak is dat op de VAX voor EOF een EOLN kan staan, waardoor EOF te laat wordt opgemerkt. Turbo-Pascal heeft in geen geval een EOLN vlak voor een EOF staan.

Door eerst te checken op EOLN, en in het geval dat waar is een READLN op de file te doen, kan deze fout voorkomen worden. En wordt EOF tijdig opgemerkt.

Bij het schrijven van een file in Cornell Condensed Format door VG-Trans trad een andere fout op. Om onduidelijke redenen liep dit steeds weer fout op het schrijven van een te lange regel. Dit is verholpen door het aantal gegevensparen dat op 1 regel geschreven wordt van 10 terug te brengen naar 9.

Een andere bug die bij het schrijven van een Cornell Condensed file optrad was het op de verkeerde plaats schrijven van opnamenummer 0000 door de computer. Het opnamenummer werd n.l. niet op het begin van een nieuwe regel geschreven, maar achter de laatste soort/abundantie-combinatie van de laatste opname. Hierdoor werd het einde van de opnamegegevens niet gesignaleerd en probeerde de computer de soortnamen ook als opnames probeerde in te lezen.

Opgemerkt kan hierbij nog worden dat VG-Trans bij het inlezen van Cornell Condensed files hetzelfde leesframe hanteert.

De FORTRAN-formaat regel wordt niet door VG-Trans geanalyseerd. Als gevolg hiervan is het moeilijk om om Cornell Condensed files van "vreemde" oorsprong te verwerken. Het is daarom aanbevelenswaardig een verbetering in VG-Trans aan te brengen, waardoor het mogelijk wordt de formaatregel te analyseren tijdens bewerking van Cornell Condensed files.

3.2.4. Keuzebepaling op basis van strings.

Veel input is variabel van grootte. De rang loopt van een return, via een getal tot volledige strings. Vooral het geval waarin een string wordt gelezen, maar de string leeg blijft, vereist op de VAX enig extra programmeerwerk t.o.v. de P.C..

Na het inlezen is moet worden nagegaan wat de bedoeling is van de gebruiker. Vaak wordt wanneer het eerste element van een string een spatie is, voor een versteks- of defaultwaarde wordt gekozen. Alleen een lege string heeft op de VAX geen lengte en is niet met spaties gevuld (op de PC heeft een lege string ook een lengte nul maar is "gevuld" met spaties). Het is daarom nodig om een lege string te vervangen door een string met één of meerdere spaties. Daarna kan verder gecheckt worden.

Een mogelijke constructie, die goed werkt, ziet er als volgt uit:

```
READLN(koord);
if koord='' then koord:='      ';
if koord[1]=' ' then etc.
```

Wat er gebeurt is dat een lege string (') vervangen wordt door een string die bestaat uit een aantal spaties.

Deze constructie heeft het voordeel dat de oorspronkelijke if-then reeks ongewijzigd kan blijven. Het onderstreepte deel is wat er voor de VAX is toegevoegd.

3.2.5. Filenamen

De meeste vragen die gesteld worden, zijn vragen waar de gebruiker gevraagd wordt een filenaam in te vullen. Dit is alleen nodig indien de filenaam afwijkt van de default. Is de naam gelijk aan de default dan kan met een return worden volstaan.

Wanneer voor een andere filenaam gekozen is, moet er rekening mee worden gehouden dat de string die beschikbaar is voor bestanden in deze versie van VG-pack 10 characters lang is.

De beschikbare ruimte is voldoende voor b.v.

- 1) xxxxxx.yyy
- 2) xxxxxx.yy
- 3) xxxxxxxx.y
- 4) xxxxxxxxxxx

Ad 4) Wanneer voor deze vorm wordt gekozen, ook bij een kortere naam, d.w.z. géén extensie, dan zal VAX-Pascal de file toch van een extensie voorzien n.l. een ".dat"-extensie.

Als gevolg van deze beperkte stringruimte binnen VG-Pack zijn de mogelijkheden voor het benoemen van datafiles beperkter dan VAX/VMS toelaat.

De VAX staat langere bestanden toe dan binnen VG-Pack mogelijk is n.l. maximaal 39 tekens. Waarbij geen onderscheid wordt gemaakt tussen hoofdletters en kleine letters; de toegestane tekens zijn a-z, 0-9, \$ en _. Daarnaast kan een filenaam nog voorzien worden van gegevens omtrent de Disk waarop, en de directory waarin het file te vinden is. Een voorbeeld van een filenaam voorzien van een directory beschrijving is :

```
[jtjvdw.vegdata]vegetatie.txt
```

N.B. De disk- en directory-gegevens tellen niet mee in de 39

beschikbare posities in een filenaam. In het voorbeeld hiervoor is de filenaam 13 posities lang. (R.C. R.U.G. 1986)

Binnen VG-Pack moeten extra gegevens (b.v. directory) wel binnen de beschikbare stringlengte (nu 10 posities) passen. Door gebruik te maken van een ".COM"-file, dit is een file waarin een reeks of batch commando's achter elkaar zijn opgenomen, kan automatisch van directory gewisseld worden en kan er voor gezorgd worden dat programma's die in een ander directory staan toch makkelijk toegankelijk zijn. Een voorbeeld van zo'n ".COM"-file is VGPREP.COM, dat ook deel uitmaakt van VG-Pack. Voor een omschrijving van dit file zie Hoofdstuk 4.

3.2.6. Omgang met en dimensionering van arrays.

Arrays kunnen op de VAX veel groter zijn dan op de Cyber of een PC. De redenen hiervoor zijn meerdere. De grootte van het gebied tussen MAXINT en -MAXINT is de grote ruimtewinner t.o.v. de P.C. Het virtueel geheugen is de oorzaak van de winst ten opzichte van de Cyber. Virtueel geheugen betekent dat de computer, een stuk extern (schijf-)geheugen gebruikt als of het RAM is. Dit betekent dat b.v. grote arrays, slechts gedeeltelijk in het werkgeheugen aanwezig zijn. De rest staat op schijf. Wanneer, tijdens de programma-executie, het deel dat in het werkgeheugen staat is verwerkt, wordt dat deel veilig weggezet op schijf. En wordt een volgend deel van schijf het werkgeheugen ingehaald. De computer regelt zelf dit schuiven met gegevens. Een zelfde effect kan door een programmeur bereikt worden op een PC of op de Cyber, door tussenuitkomsten etc. in tijdelijke files op schijf op te bergen. Deze oplossing is vaak tijdrovend, door relatief trage toegang tot externe geheugens. Door het bedrijfssysteem van de computer met snelle methodes voor dergelijke operatie uit te breiden kan veel tijd bespaard worden.

De programmeur die op de VAX grote arrays gebruikt moet wel rekening houden met de manier waarop de VAX met het virtuele geheugen omgaat. Hij/zij dient erop toe te zien dat de eerste index van het array het langzaamst oploopt, en dat eventuele hogere indices sneller oplopen. Hoe hoger de array-dimensie die door de index wordt aangeduid, des te sneller dient de waarde van deze index toe te nemen ten opzichte van de andere indices. Wordt hier geen rekening meegehouden, dan wordt het array door het programma haaks op de werking van de VAX doorgewerkt. Het gevolg is dat de snelheid van het programma drastisch afneemt, wegens gigantische wisselactiviteiten tussen werk- en virtueel geheugen.

Een foutmelding die met het dimensioneren van arrays verband houdt is: **Exceeded page file quota**. Deze melding betekent dat het programma meer virtueel geheugen vergt dan beschikbaar is. Declaraties van arrays zullen in dit geval bescheidener opgezet moeten worden. Dezelfde foutmelding zal vermoedelijk ook optreden wanneer een programma dat met pointers werkt (dynamische variabelen) een te groot gegevensbestand krijgt aangeboden.

3.3. Suggesties ter verbetering van VG-Pack.

Om het pakket nog meer verwantschap te geven met VEGROW is het gebruik maken van nieuwe mogelijkheden voor P.C.-achtigere programma's aan te bevelen. Een voorbeeld in dit kader zou kunnen zijn het wissen van het scherm, m.b.v. een procedure van Maarten van Gelderen. (Hiervoor moet een 'symbol' worden gedeclareerd n.l. `termtype==tvi912`. Dit is om de juiste schermaansturingscodes te kunnen kiezen die bij de gebruikte terminal horen. Een programma kan dan met meerdere types terminal correct werken.) Waarbij opgemerkt moet worden dat de Beehive-terminal niet ondersteund wordt. Dit oude type zal in de toekomst waarschijnlijk niet beter ondersteund worden. Wanneer bij het gebruik van een Beehive-terminal het terminaltype als TVIxxx is ingesteld dan werkt de ClrScr-procedure niet correct, maar levert verder geen problemen op. De teksten worden dan normaal onder elkaar op het scherm geschreven, waarbij de tekst aan de bovenkant uit beeld verdwijnt. Vanwege het streven om de verwantschap tussen de programma's op de P.C. en de VAX8600 is het aan te raden, met het voorgaande in gedachten, nieuwere P.C.-programma's aan te passen aan de VAX. En deze te gebruiken ter vervanging van de versie die op dat moment in VG-Pack is opgenomen.

Aan te raden is ook uitbreiding van het pakket met een programma (of een uitbreiding van Trans) zodat Cornell Condensed gegevensbestanden gelezen worden volgens het formaat dat in het file zelf omschreven is op regel 2. Vooral de mogelijkheid die dan ontstaat om Cornell Condensed bestanden van vreemde oorsprong te verwerken is nuttig, met name doordat van deze bestanden geen equivalenten in een van de andere formaten van VG-Pack zullen bestaan.

Meer ruimte voor de gebruiker om de files te kunnen benoemen is iets dat zeker het gebruiksgemak zal bevorderen.

Uitbreiding van VG-Pack met Select, maakt het mogelijk om op basis van tijdens de executie te bepalen critereria opnames uit een file in Databank-formaat te selecteren. De geselecteerde files worden naar een nieuw file gecopiëerd. Door deze faciliteit aan VG-Pack toe te voegen ontstaat de mogelijkheid om op handige wijze de samenstelling van datasets te manipuleren.

Andere uitbreidingen voor VG-Pack die de bruikbaarheid van het pakket kunnen verhogen zijn: het opnemen in het pakket van een programma voor bestandsbeheer. Voor kopiëren, verwijderen, herbenoemen en bekijken van bestanden. Opnemen van een automatische aanroep van een bestaande editor of een speciale editor in het pakket.

3.4. Van Turbo naar VAX-Pascal.

Hoe te handelen bij het overnemen en opnieuw implementeren van Turbo-Pascal programma's van de PC-versie VEGROW op de VAX8600. De meeste P.C.-programma's zijn verder geëvolueerd dan de programma's op de VAX. En sommige programma's zijn grotendeels opnieuw geprogrammeerd, waardoor ze een stuk sneller zijn geworden.

Wanneer een programma aan de VAX wordt aangepast, is de moeite waard om GoToXY en andere Turbo-trekjes, indien mogelijk te behouden. Door gebruik te maken van (nieuwe) procedures van Maarten van Gelderen is dit mogelijk.

Deze procedures, die hetzelfde doen en dezelfde naam hebben als hun Turbo-Pascal-equivalenten, kunnen voor het merendeel op dezelfde wijze worden aangeropen. De aanpassingen aan het programma zijn daardoor minimaal. Bij gebruik van Turbotrekjes op de VAX, voor procedures die in Turbo "predeclared" zijn en die een "Van Gelderen"-vervanger hebben, moet een declaratie als external in de source-tekst van de programma's worden opgenomen.

De makkelijkste methode om de juiste declaraties in het file op te nemen is: opnemen van de file 'pasproc.hdr' in het begin van de programmatekst.

Wanneer slechts enkele van deze routines worden gebruikt kunnen ook uitsluitend de benodigde declaraties worden opgenomen. Door ofwel zelf in typen van die declaraties, of de overbodige declaraties uit pasproc.hdr te verwijderen, na opnemen in de brontekst.

Vervolgens moet het programma worden nagelopen op:

- declaraties van strings,
- het inlezen van strings van input,
- het lezen van gegevens van files etc.

Waarbij de voorgaande paragrafen als handleiding kunnen dienen.

Tot slot compileren met de opdracht:

pascal programma

en vervolgens linken met de runtime library en de al gecompileerde procedures:

link programma + disk\$1:[gurv.maarten]pasproc/lib

Hierna kan het programma gebruikt worden. De opdracht om een programma te starten is:

run programma

In de voorgaande voorbeelden is er vanuit gegaan dat de files een extensie hadden die overeenkomt met de default.

D.w.z. voor een te compileren programma: **.pas**,

voor een te linken programma: **.obj**

en voor een evt. libraryfile die tijdens het linken moet worden gebruikt: **.olb**.

Terwijl het bruikbare programma dan een **.exe**-extensie heeft.

Alle extensies van filenamen zijn niet dwingend, echter wanneer van de conventie wordt afgeweken, dan moeten de extensies volledig en correct worden ingevoerd op de commando-regel.

4. Omgang met de VAX8600 als computersysteem.

4.1. Handig gebruik van de VAX.

Het gebruik van de VAX kan, door de gebruiker, sterk vereenvoudigd worden (R.C. R.U.G., 1986). De middelen die hiertoe tot zijn/haar beschikking staan zijn o.a.:

- Logicals - die gebruikt kunnen worden om vaak voorkomende filenamen en namen van subdirectories te vervangen.
- Symbols - die dienen om commando's af te korten, en die tevens de mogelijkheid bieden om een commando, steeds op een door de gebruiker gekozen wijze te laten uitvoeren, zonder keer op keer opnieuw alle parameters en qualificiers te moeten intypen.
- Comfiles - dit zijn files, die volgens de naamgevingsconventie op de VAX een filenaam hebben met de volgende structuur: XXXXX.com. In deze files kunnen reeksen commando's samengevoegd worden en als een groep in één keer worden uitgevoerd. Het is ook mogelijk om bepaalde commando's alleen uit te laten voeren wanneer aan een bepaalde conditie is voldaan. Ook kan met enig kunst- en vliegwerk, vanuit zo'n file een interactief programma gestart worden. Een com-file is vergelijkbaar in functie met het batchfile (.bat-extensie) zoals de P.C. kent en vertoont ook duidelijke gelijkenis met 'procfil' op de Cyber 170.

Om een comfile uit te laten voeren moet een commando als het volgende worden ingetikt:

@voorbeeld

in dit geval heet het file 'voorbeeld.com'.

@voorbeeld.doe

het file heet in dit geval 'voorbeeld.doe',

andere extensies dan .com zijn wel mogelijk, maar moeten dan bij het starten nadrukkelijk worden opgegeven.

Een voorbeeld van een veel voorkomende comfile is: login.com. De commando's die in dit file staan worden automatisch uitgevoerd tijdens de inlogprocedure. Heel handig om elke keer opnieuw de beschikking te hebben over dezelfde logicals en symbols. Op deze manier kan men erop blindvaren dat de computer steeds op dezelfde manier op een commando zal reageren. Login.com heeft eenzelfde functie als 'autoexec.bat' van de P.C..

Een ander voorbeeld is vgprep.com. In dit file wordt onder andere van conditionele uitvoering van commando's gebruik gemaakt.

Figuur 1: VGPREP.com

```
$ ! Redefinition of the systemprompt to remind you that
$ ! than you are free to use VG-pack facilities
$ set prompt="VG-pack >> "
$ ! The directory specification in the following logical
$ ! should match the directory where the
$ ! VG-pack files are to be found on your system.
$ assign [gwbb.jtjvdw] vgexe
$ ! The next logical is for the directory containing your
```

VG-Pack Programmeurs Manual

```
$ ! datafiles. Preferably this is another directory than
$ ! the one containing the program-files of VG-pack.
$ ! In this way the files do not clutter up the transparency
$ ! of your data- or program-directory
$ assign [gwbb.jtjvdw.vegdata] data
$ ! This is to make sure that VG-pack is started in your
$ ! data directory enabling it to find your data.
$ set default data
$ ! Symbol to start VG-pack with.
$ vgpac := run vgscreen
$ ! The next two lines make a file easily accessible. This file
$ ! reduces system overhead, by deleting the helpfiles
$ ! and tabel.veg from your datadirectory. If this file finds
$ ! that these files were not present there when VG-pack was
$ ! activated for the last time.
$ end := @vgend
$ assign vgexe:vgend vgend
$ ! vgexe in the next command should be replaced with data
$ ! when vgprep.com is in your data-directory in stead of
$ ! in the VG-pack-directory
$ assign vgexe:vgprep vgprep
$ ! The next command enable VG-pack to execute its
$ ! programs at your request with these programs being
$ ! present in your own datadirectory. Thus reducing system
$ ! overhead.
$ ASSIGN vgexe:aclus aclus
$ ASSIGN vgexe:assan assan
```

Hier staan in werkelijkheid nog meer assign-statements.

```
$ ASSIGN vgexe:vgmain vgmain
$ ASSIGN vgexe:vgscreen vgscreen
$ ! The next construction determines if the help-files
$ ! and tabel.veg are already present in your data-
$ ! directory. (This is where they need to be if you want
$ ! to be able to use the help-facility)
$ ! If these files are not yet present, then they are copied
$ ! to your data directory. In other cases they are not copied
$ ! which saves you some time.
$ ! If the files were not present, than you can get rid of them
$ ! by typing "end" when you are done with VG-pack
$ Hulpexist == f$search ("data:*.hlp;1")
$ Tabelexist == f$search ("data:tabel.veg;1")
$ if hulpexist.eqs. "" then copy vgexe:*.hlp data:*.hlp;1
$ if tabelexist.eqs. "" then copy vgexe:tabel.veg data:tabel.-
veg;1
```

In het voorgaande com-file zijn een aantal truc's gebruikt, die nog enige nadere uitleg krijgen. VAX/VMS maakt in commando's geen onderscheid tussen hoofd- en kleine letters, net als MS/PC-DOS.

4.1.1. Conditionele uitvoering van één of meerdere opdrachten.

Bij gebruik van het "Lexical Element" f\$search kan een constructie gemaakt worden, die qua functionaliteit vergelijkbaar is met de 'if exist file'-functie van MS/PC-dos.

`f$search` ("volledige filenaam evt. met wildcards"), resulteert in een lege string indien er geen file(s) worden gevonden die aan het zoekcriterium voldoen. In alle andere gevallen krijgt `f$search` een string-waarde overeenkomend met de volledige filenaam van het gezochte file. Wanneer van wildcards of jokers gebruik is gemaakt, dan is er een string met filenaam indien er minstens 1 file, dat aan het zoekcriterium voldoet, is gevonden. Door het resultaat van `f$search` aan een symbol toe te kennen kan het resultaat worden bewaard. Bijvoorbeeld kan het resultaat later gebruikt worden om een bepaalde actie al dan niet uit te laten voeren, wanneer het symbol gelijk is aan (.eqs.) of juist niet gelijk is aan (.nes.) iets anders. Zie b.v. de laatste vijf regels van Figuur 1: VGPREP.com hiervoor.

Andere voorbeelden van commando's die beschikbaar zijn binnen de 'f\$'-reeks zijn: `f$time()` en `f$user`. De eerste kan b.v. gebruikt worden in comfiles om de tijd van een actie vast te leggen. De tweede geeft als resultaat de UIC (= User Identification Code), deze UIC is nodig bij sommige commando's.

4.1.2. Interactieve aanroep van programma's vanuit com-files.

Programma's die vanuit een com-file worden gestart verwachten ook de invoer vanuit dat file. Om vanaf de terminal een uit com-file gestart programma te kunnen beïnvloeden moeten de volgende regels in het com-file worden opgenomen:

```
define /user sys$input sys$command  
run programma
```

Een volgend programma dat vanuit com-file wordt aangeroepen, moet om interactief te kunnen worden gebruikt op dezelfde wijze worden gestart. De define- en run-statement moeten direct op elkaar volgen. En moeten voor elk programma, dat op deze wijze gestart moet worden, weer op deze wijze herhaald worden.

4.1.3. Gegevensbescherming.

Op de VAX is het mogelijk om eigen files te beschermen. Dit wordt gedaan door de vier groepen gebruikers die het systeem kent aan te geven wat voor die groep is toegestaan.

De vier groepen die het systeem kent, voor onder andere het beschermen van bestanden, zijn :

User: de eigenaar van de files,

Group: degenen die samen met de User een hoofddirectorie delen. (Voor PlantenOecologie, alle Users die bij GWBB horen.)

World: Iedereen.

System: een speciaal stukje wereld. Dat zich met name van de rest van de wereld onderscheid door de privileges die het bezit.

Opgemerkt moet worden, dat wanneer een actie voor een grote groep, met lage prioriteit, is toegestaan, daardoor de kleinere groepen, met hogere prioriteit, die er binnen vallen deze actie ook mogen uitvoeren. Dat betekent dat wanneer aan de groep "WORLD" iets wordt toegestaan, ook echt iedereen dat mag doen. Er is dan geen mogelijkheid om b.v. SYSTEM te beletten deze actie uit te voeren.

Het commando om de protectiecode in te stellen is:

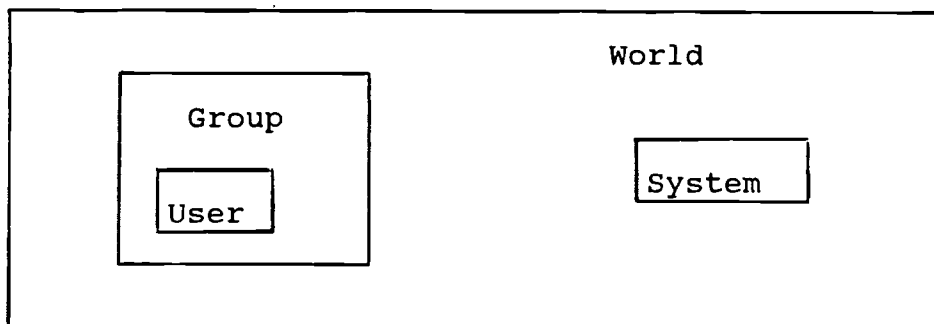
Set protection=(System:r,Owner:rwed,Group:re) /default

deze vorm van het commando stelt de protectie in voor alle files die tijdens dezelfde sessie nog worden gemaakt, of totdat er een andere protectiecode wordt ingesteld. De wijze waarop reeds bestaande files worden beschermd wordt met dit commando niet gewijzigd.

Set protection=(S,O,G,W:RWED) filenaam

wijzigt de protectie voor de genoemde reeds bestaande files.

Figuur 2: FileProtectieGroepen op de VAX 8600



Betekenis van de letters in de protectiecode:

R = READ (de files kunnen gelezen worden)

W = WRITE (de files kunnen gewijzigd worden)

E = EXECUTE (programma's kunnen ge'run'd worden)

D = DELETE (files mogen gewist worden).

Wat de instelling van de filebescherming is, is zichtbaar te maken met: "SHOW PROTECTION"

en door het commando "DIRECTORY/PROTECTION filenaam" is te zien wat de protectiecode is van een file. Wanneer geen filenaam wordt opgegeven wordt een overzicht gegeven van alle files, wanneer een filenaam met jokers (wild cards) wordt opgegeven wordt van alle passende files die passen de naam en de protectie op het beeldscherm getoond.

4.1.4. Terminal

Tijdens het werken met de VAX is het verstandig er voor zorg te dragen dat de computer nauwkeurig weet welk type terminal de gebruiker gebruik. De reden is dat terminals van elkaar verschillen in schermaansturing en in sommige andere mogelijkheden.

Met name bij gebruik van een DEC VT100 of een PC die zo'n terminal emuleert (nabootst) is het heel handig om dit aan de VAX mee te delen. Bij dit type terminal bestaat de mogelijkheid om de laatste twintig commando's te laten onthouden. Deze commando's kunnen weer opgeroepen worden door met de op- en neer-pijltjes weer te zoeken. Ook is het mogelijk om een opnieuw opgeroepen commando te wijzigen (editen) alvorens de computer opdracht te

geven het commando uit te voeren. Zie voor het editen van commando's RC-publicatie 13 "Inleiding VAX/VMS".

Het commando om de VAX te laten inspelen op de aanwezigheid van een VT100 of gelijkwaardige terminal is:

Set Terminal /dec_crt

Andere instellingsmogelijkheden zijn er vele. Door het commando **show terminal** is te zien wat er ingesteld kan worden, en hoe de instelling op dat moment is. **Help set terminal** is het commando dat hulp aanvraagt over het instellen van terminal-parameters.

4.1.5. Hulpfaciliteiten.

De VAX 8600 heeft een uitgebreide on-line hulpverlening. Indien een probleem zich voor doet, kan een overzicht worden verkregen van hoofdonderwerpen waarover hulp beschikbaar is door **HELP** achter de systeemprompt in te tikken. Achter elk hoofdonderwerp (topic) gaan andere deelonderwerpen (sub-topics) schuil. Ook binnen een aantal toepassingsprogramma's zoals editors, is hulp beschikbaar.

Door gebruik te maken van deze hulpfaciliteiten kunnen een boel problemen achter de terminal worden opgelost. Bedenk wel dat een beeldscherm minder makkelijk leest en zoekt dan een stapel papieren. Met name bij meer complexe problemen loont het de moeite loont om na korte tijd vruchteloos zoeken in de computer bij het gedrukte woord te rade te gaan.

Wanneer bij het optreden van een probleem direct bekend is dat de oplossing gezocht moet worden bij een sub-sub-sub-topic kan direct worden door gekozen naar dat onderdeel. Dit levert een niet onaanzienlijke tijdwinst op doordat niet steeds op het schrijven van tussenliggende teksten gewacht hoeft te worden. Voorbeelden van dit "rechtstreeks doorkiezen" zijn:

Help pascal

Help pascal predeclared

en ook het voorbeeld over help aan het eind van paragraaf 4.1.4 Terminal.

4.2. Omgang met de Vax8600.

Voor een vlotte kennismaking met de VAX is het nodig om enige kennis te hebben van een aantal commando's. De commando's, die hier worden genoemd, zijn allemaal ook terug te vinden in Inleiding VAX/VMS (RC-publicatie 13, R.C. R.U.G., 1986). Deze en andere commando's worden daar uitgebreider behandeld. De commando's die hier in het kort zullen worden behandeld hebben allen te maken met het beheer van bestanden. Ze maken het mogelijk de volgende functies uit te voeren:

- Overzicht van de aanwezige bestanden.
- Mogelijkheid tot het kopiëren/verwijderen/herbenoemen van bestanden.
- Mogelijkheid om bestanden te wijzigen (editen).
- Wijzigen van het Password.

Van de commando's worden enkele nuttige varianten behandeld.

4.2.1. Overzicht Bestanden

VG-Pack Programmeurs Manual

Het commando om een overzicht van de aanwezige bestanden te krijgen is:

DIR

Nuttige toevoegingen aan dit commando kunnen zijn :

Alleen files met een bepaalde naam/extensie: b.v. **DIR *.veg**
geeft een overzicht van alle (*) files met .veg als extensie

Laat ook de filebeschermingscode zien: **DIR /PROT**

Laat de filegrootte zien (per blok van 512 bytes): **DIR /SIZE**

Deze mogelijkheden kunnen ook gecombineerd worden.

4.2.2. Copiëren en verwijderen van Bestanden

Het commando om een bestand te kopiëren is:

Copy oudenaam.ext nieuvenaam.ext

Indien de filenamen worden weggelaten vraagt de computer eerst:

From:

en vervolgens:

To:

Waar de filenamen alsnog kunnen worden ingevuld.

Om een bestand te verwijderen is het volgende commando beschikbaar:

DELETE /CONFIRM filenaam.ext;nr

Door dit commando te gebruiken vraagt de computer om een bevestiging (confirmation) alvorens het file daadwerkelijk te verwijderen. Het versienummer van het file moet worden opgegeven, wanneer dit wordt weggelaten volgt een mededeling van de computer.

DEL /CONFIRM opnames82?????.dat;*

Dit commando verwijdert, met bevestiging per file, alle files met opnamegegevens uit 1982, voor alle (;*) versies en waarbij de 4 vraagtekens een 4tekencode zou kunnen zijn voor de plaats waar de opnames gemaakt zijn.

Indien hier ook een * was gebruikt zouden ook files met een langere plaatscode o.i.d. meegenomen worden.

N.B. voor deze zogeheten 'wild cards' (=jokers, ? en *) geldt dat ze resp. staan voor 1 of geen willekeurig teken, en voor 0-X tekens.

Om een verwijdering te bevestigen moet 'y' of 'Y' + RETURN worden ingetikt. Voor behouden van een file is een druk RETURN voldoende; 'N' of 'n' is n.l. het voorgeprogrammeerde antwoord. Deze tekens mogen echter ook worden gebruikt.

Een ander commando waarmee bestanden verwijderd kunnen worden is:

PURGE filenaam.ext

Dit commando verwijderd alle bestanden die met de opgegeven naamsomschrijving overeenstemmen, maar laat de versie met het hoogste versienummer bestaan. Evt. kan het commando zo worden aangepast dat de laatste zoveel versies behouden blijven.

Bestanden kunnen worden herbenoemd, d.w.z. de naam waaronder de gegevens bereikbaar zijn kan gewijzigd worden. Het commando hiervoor is:

RENAME oudenaam.ext nieuwnaam.ext

Indien de filenamen worden weggelaten vraagt de computer eerst:

From:

en vervolgens:

To:

Waar de filenamen alsnog kunnen worden ingevuld.

4.2.3. Wijzigen van bestanden

Het wijzigen of editen van bestanden is op de VAX met verschillende editors mogelijk. Beschikbaar zijn o.a. EDIT, EDN en ESE. De eerste is een bij de computer horende regeeditor. De tweede is een VAX-versie van de EDinburgh editor, deze verschilt in enige opzichten van de CYBER-versie. Het is ook een regeeditor.

ESE, is een schermeditor, vergelijkbaar met veel editors op P.C.'s. Met de cursor en de pijltjes-toetsen kan de plaats van de fout worden aangewezen/bereikt, en kan de verbetering direct plaatsvinden. In ESE zijn kunnen o.a. zoeken naar en vervangen van tekstfragmenten en andere zaken EDN-commando's gebruikt worden. ESE kan op de VAX gebruikt worden in combinatie met de volgende Terminals: Televideo 910,912,920,925, DEC VT100 en P.C.'s die met Kermit een terminal emuleren. Na een aanroep vraagt ESE met wat voor terminal gewerkt wordt.

Van welke editor U gebruik zal maken, zal van een aantal factoren afhangen. O.a. kent U een van de editors al, wat moet er gedaan worden, welke terminal gebruikt U.

De inhoud van een tekst- of gegevensfile kan op de VAX bekeken worden zonder gebruik te maken van een editor. Hiertoe is het **TYPE** filenaam.ext commando beschikbaar.

4.2.4. Printen van bestanden

Na het uitvoeren van analyses, moet men de resultaten ook op papier kunnen krijgen. Er zijn twee manieren beschikbaar om op de VAX8600 een file te laten printen.

De eerste mogelijkheid laat een file printen op de standaard-printer voor de VAX8600. Deze printer staat op de 1^e verdieping van het WSN-gebouw in een printerruimte.

Print filenaam

De tweede mogelijkheid laat een file afdrukken via een printer die is aangesloten op het printernetwerk. Hiermee kan een file b.v. worden afgedrukt door de printer die op de Oude Boteringestraat staat.

Spool filenaam printernaam

Voor printernaam moeten dezelfde aanduidingen worden gebruikt als bij de route-opdracht van de CYBER. Dus voor Oude Boteringestraat

is dit OB23. RCZB= zelfbedieningsstation RekenCentrum; WSN1= zelfbedieningsstation 1^e verdieping WSN.

4.2.5. Het Password

Om toegang te krijgen tot de VAX moet U in het bezit zijn van een UserName en Password, en iemand, de MasterUser, moet toestand dat U op zijn (evt. uw eigen) rekening de computer te gebruikt. Computergebruik is in principe n.l. niet gratis. Binnen de Universiteit is door veel instituten etc. een afkoopsom betaald, binnen zekere grenzen kan er onbeperkt worden gerekend voor dat geld. Andere zaken zoals het laten printen van teksten op de laserprinter of het laten plotten van grafieken kunnen echter extra geld kosten.

Uw Password heeft een beperkte geldigheidsduur n.l. 180 dagen. Na die periode moet U een nieuw password kiezen. Wanneer dit moet gebeuren geeft de computer bij het inloggen de volgende boodschap: Your password has expired. En nog wat andere tekst. In deze tekst staat al hoe het password gewijzigd kan worden. Het commando om het password te wijzigen is:

SET PASSWORD

De computer vraagt vervolgens:

Old Password:

New Password:

Verification:

De verification/controlestep wordt evt. herhaald, totdat het nieuwe password nogmaals correct d.w.z. gelijk aan de eerste keer wordt ingetypt.

Indien tijdens de inlogprocedure gemeld wordt dat het password is verlopen moet tijdens dezelfde sessie nog het password vernieuwd worden. Wordt dit niet gedaan dan is het onmogelijk om een volgende keer contact met de computer te leggen.

De enige mogelijkheid om dan opnieuw toegang tot de computer te krijgen is uw masteruser verzoeken uw Password opnieuw te initialiseren. Na initialisatie is het Password gelijk aan uw username, en moet het m.b.v. SET PASSWORD ingesteld worden.

Voorkomen is beter dan genezen, voorkom dat het password verloopt. Vervang het password vaker dan eenmaal in de 180 dagen wanneer de computer erom vraagt. Dit is tegelijkertijd een verbetering van de bescherming van uw gegevens.

4.3. Foutmeldingen op de VAX

Wanneer een computer er niet in slaagt een ontvangen opdracht succesvol uit te voeren laat het apparaat dit weten door middel van een foutmelding. Afhankelijk van de gebruikte computer en programmatuur is de kwaliteit en vorm van zo'n foutmelding erg wisselend. Soms is het slechts een nummer, wat op zich bijzonder weinig informatie geeft. Vaak is het wat meer, meestal dan een ook weinig informatieve melding als "ERROR" of "BAD COMMAND"

VG-Pack Programmeurs Manual

Figuur 3: Voorbeelden VAX-foutmeldingen

- a) een pascal programma dat vastloopt.
- b) een VAX-commando dat een bestand niet kan vinden.
- c) een melding dat een bepaald programma te veel vergt van het virtuele geheugen,

```
-----  
%PAS-F-GETAFTEOF, GET attempted after end-of-file  
File "FD" Filename"HSC000$DUA4:[GWBB.JTJVDW.VEGDATA]GANS GC.VEG;8"  
%TRACE-F-TRACEBACK, symbolic stack dump follows  
module name routine name line rel PC abs PC  
00034907 00034907  
0002FD82 0002FD82  
DIMAT READREC 246 0000017F 0000D34F  
DIMAT OPNMAT 271 0000012B 0000D617  
DIMAT DIMAT 374 0000004E 0000B686
```

```
VG-pack >> type gansploec  
%TYPE-W-SEARCHFAIL, error searching for  
HSC000$DUA4:[GWBB.JTJVDW.VEGDATA]GANS PLOEC.LIS;  
-RMS-E-FNF, file not found  
VG-pack >> type gansploec.
```

```
%DCL-W-ACTIMAGE, error activating image ASSAN  
-CLI-E-IMGNAME, image file HSC000$DUA4:[GWBB.JTJVDW]ASSAN.EXE;7  
-SYSTEM-F-EXQUOTA, exceeded quota  
-----
```

Wanneer de in Figuur 3 beschreven foutmeldingen worden bekeken is te zien dat de meldingen volgens een bepaald systeem zijn opgebouwd. Ze beginnen allemaal met een %-teken, daarna komt een meestal 3-tekens bevattende code die aangeeft waar er iets is misgegaan (PAS voor Pascal programma's; TYPE voor het type commando, DCL voor fouten in zaken aangaande Digital Command Language etc.). Vervolgens volgt tussen gedachtenstreepjes een teken (E=error, F=failure, W=warning) dat aangeeft hoe ernstig het probleem is. Na deze code volgt een kryptische beschrijving van wat er is misgegaan b.v. GETAFTEOF, echter hierna volgt een langere en betere omschrijving van hetgeen er is misgegaan. In dit geval is getracht te lezen (GET) voorbij het einde van het file, waarbij ook wordt aangegeven met welk file het is misgegaan. Bij Pascal en Fortran programma's wordt tevens met een TRACEBACK aangegeven waar (regel en procedure) de fout is opgetreden.

Treedt er tijdens het gebruik van de VAX8600 een foutmelding op dan is er geen reden tot paniek. Alle bestanden zijn nog intact en bruikbaar. Er wordt veel ruimte genomen voor de foutmelding, maar de foutmelding bevat ook veel informatie. B.V. wanneer bij de tweede foutmelding uit Figuur 3 uitsluitend "FILE NOT FOUND" zou worden gemeld, zou een volslagen onduidelijke situatie ontstaan. Immers het file is er wel degelijk. Het probleem is dat de computer het commando anders interpreteert dan verwacht. Er wordt gezocht naar een file met ".LIS"-extensie. Door hetzelfde

VG-Pack Programmeurs Manual

commando nogmaals in te voeren, met toevoeging van een "." aan het einde van de filenaam lukt het wel.

Type "filenaam" : betekent kennelijk type "filenaam.lis"

Type "filenaam.ext" : is duidelijk.

Type "filenaam." : is de vorm die nodig is om een extensieloos file op het beeldscherm afgebeeld te krijgen.

5. Literatuurlijst.

Braak C.J.F. Ter (1987a)

Canoco - a FORTRAN program for canonical community ordination by [partial][detrended] correspondance analysis, principal component analysis and redundancy analysis (version 2.1).

IWIS-TNO, Wageningen

Braak C.J.F. Ter (1985)

Correspondance Analysis of Incidence and Abundance Data: Properties in Terms of a Unimodal Response Model.

IWIS-TNO, Wageningen, report

Braak C.J.F. Ter (1987b)

The analysis of vegetation-environment relationships by canonical correspondance analysis

Vegetatio 69 (in press)

Braak C.J.F. ter, Looman, C.W.N. (1987)

Weighted averaging, logistic regression and the Gaussian response model.

Vegetatio (in press)

Braak C.J.F., Colin Prentice, I. (1986)

A theory of gradient analysis

ITI-TNO Wageningen

Everitt B. (1974)

Cluster Analysis

SSRC, Heinemann Educ. Books Ltd.

Gauch H.G. jr. (1982)

Multivariate analysis in community ecology

Cambridge University Press

Heiser Willem J. (1983)

Joint Ordination of Species and Sites: The Unfolding Technique.

New Dev.s in Num. Ecol. ed. P. Legendre & L. Legendre

Springer Verlag, Berlin

Hill M.O. (1973)

Reciprocal Averaging: an eigenvector method of ordination.

J. of Ecology, 63, 237-250

Hill M.O., Gauch H.G. jr. (1980)

Detrended Correspondance Analysis: an improved ordination technique.

Vegetatio 41, 47-58

Kershaw K.A., Looney J.H.H. (1985)

Quantative and Dynamic Plant Ecology (3.ed)

Edward Arnolds Lim., London

VG-Pack Programmeurs Manual

Kramer, Koen (1986)

Evaluatie van een aantal hiërarchisch-polythetisch-divisie
clustermethoden met betrekking tot de verwerking van vegetatie??
doctoraal verslag, R.U.G.

Legendre L. & Legendre P. (1983)

Numerical Ecology.

Elsevier Scientific Publ. Comp., Amsterdam

Lindeman R.H., Merenda P.F., Gold, R.Z. (1980)

Introduction to multivariate analysis

Scott, Foresman & Company

Maarel, E. van der (1982)

On the manipulation and editing of phytosociological and ecological data.

Vegetatio 50, 71-76

Orloci L. (1975)

Multivariate analysis in vegetation research.

Junk, Den Haag

R.C. R.U.G. (1986)

Inleiding VAX/VMS

RC-publikatie 13

Wal, J.T. van der (1987)

Canonische Correspondentie Analyse

doctoraal verslag, R.U. Groningen

Wal, J.T. van der (1987)

VG-Pack Manual

handleiding, R.U. Groningen

Whittaker R. (ed.), Tuxen R. (ed.) (1973)

Handbook of Vegetation Science (V) Ordination and Classification
of Communities

Junk, Den Haag

Wiegand, Gerhard (1986)

Grenzen und Möglichkeiten der Datenanalyse in der Pflanzenökologie.

Tuexenia 6, 365-377

Wildi, Otto (1986)

Konzeption vegetationskundlicher Untersuchungen am Beispiel einer
Gradientenanalyse.

Tuexenia 6, 379-396

Wishart D. (1978)

Clustan, User Manual (3 ed.)

Edinburgh University

VG-Pack Programmeurs Manual

Lijst van Figuren.

Figuur 1: VGPREP.com (10)

Figuur 2: FileProtectieGroepen op de VAX 8600 (13)

Figuur 3: Voorbeelden VAX-foutmeldingen (18)

VG-Pack Programmeurs Manual

Lijst van Tabellen.

Tabel 1: Verschil in maximale integer tussen P.C. en VAX. (2)

Index

Arrays (7)
 BATCH (3)
 Bestandsbeheer (8), (14)
 ClrScr (3)
 Comfiles (10)
 interactieve programma-aanroep (12)
 login.com (10)
 vgprep.com (7), (10)
 Commando
 .eqs. (12)
 .nes. (12)
 @voorbeeld (10)
 Copy (15)
 define /user sys\$input sys\$command run programma (12)
 DELETE (15)
 DIR (15)
 DIRECTORY/PROTECTION filenaam (13)
 f\$search ("volledige filenaam evt. met wildcards") (12)
 f\$time (12)
 f\$user (12)
 HELP (14)
 Help set terminal (14)
 link programma + [gurv.maarten]pasproc/lib (4), (9)
 pascal programma (4), (9)
 Print (16)
 PURGE (15)
 RENAME (16)
 run programma (4), (9)
 SET PASSWORD (17)
 Set protection=(S,O,G,W:RWED) filenaam (13)
 Set protection=(System:r,Owner:rwed,Group:re) /default
 (13)
 Set Terminal /dec crt (14)
 SHOW PROTECTION (13)
 show terminal (14)
 Spool (16)
 TYPE filenaam.ext (16)
 Cornell Condensed Format (5), (8)
 disk\$: (3)
 Editors (16)
 EDIT (16)
 EDN (16)
 ESE (16)
 Extensies (9)
 External (9)
 Filenaam (6)
 beschikbare ruimte (6)
 VAX/VMS-filenames (6)
 Fortran (1)
 formaatregel (5)
 Foutmelding (17)
 Exceeded page file quota (7)
 Your password has expired (17)
 Gegevensbescherming (12)

VG-Pack Programmeurs Manual

- FileProtectieGroepen (13)
 - Group (12)
 - protectiecode (13)
 - System (12)
 - User (12)
 - World (12)
- Help (14)
- Inlezen gegevens uit file (5)
- Interactief (3)
- Library (3)
- Logicals (10)
- MasterUser (17)
- MAXINT (2), (7)
- NextCommand (3)
- Pascal (1), (2), (3), (4), (6), (9)
- Password (14), (17)
 - geldigheidsduur (17)
- Strings (4)
 - declaraties van - (9)
 - inlezen van - (9)
 - keuzebepaling op basis van - (6)
 - lezen van strings (5)
- Symbols (10)
- Terminal (13), (16)
- Termtyp (3), (8)
- Turbo-Pascal (2), (3), (4), (5), (9)
- UserName (17)
- VAX8600 (1)
- VEGROW (2), (8), (9)
- VG-Pack (1), (2), (4), (6), (7), (8)
- Virtueel geheugen (3), (7)
- WaitForReturn (3)

