

**stichting
mathematisch
centrum**



AFDELING INFORMATICA

IN 11/76

SEPTEMBER

P.J.W. TEN HAGEN & R. VAN VLIET

TEKSTVERWERKING

2e boerhaavestraat 49 amsterdam

BIBLIOTHEEK MATHEMATISCH CENTRUM
—AMSTERDAM—

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

Tekstverwerking

door

P.J.W. ten Hagen & R. van Vliet

0. Inleiding.

In dit rapport wordt een beschrijving gegeven van een tekstverwerkend systeem, dat op het ogenblik op het Mathematisch Centrum wordt ontwikkeld. Achtereenvolgens komen aan de orde:

- De functies die het systeem voor gebruikers binnen en buiten het MC moet kunnen vervullen.
- De middelen (apparatuur en programmatuur) waarmee het systeem gemaakt wordt.
- De manier waarop het gemaakt zal worden en een schatting van de inspanning en tijd die de programmering van het systeem zullen vergen.

In deze inleiding zal getracht worden de belangrijkste aspecten van een tekstverwerkend systeem toe te lichten, eventueel aan de hand van voorbeelden uit eigen bron of uit de literatuur.

0.1 Een Bedrijfssysteem voor Tekstverwerking.

Een bedrijfssysteem voor tekstverwerking moet in staat zijn de gebruiker te helpen bij het invoeren en corrigeren van zijn tekst (editor), vervolgens moet de tekst opgeslagen kunnen worden (files), de tekst moet door de beschikbare programma's geanalyseerd en/of getransformeerd kunnen worden (utilities) en tenslotte moet het systeem de teksten weer in de gewenste vorm kunnen afdrukken (tekstproductie).

In een bedrijfssysteem voor de verwerking van programma's zijn alle faciliteiten, nodig voor tekstverwerking, in principe reeds aanwezig. De preparatie van een gewoon computer programma doorloopt nl. dezelfde fasen als hierboven geschetst voor tekstverwerking. Het hoeft ons dan ook niet te verbazen dat alle rekencentra hun computergebruikers wel een aantal meer of minder geavanceerde hulpprogramma's voor tekstbewerking kunnen aanbieden. Hulpprogramma's, die met weinig inspanning zijn verkregen door aanpassing van bestaande programmatuur.

Een tekstverwerkend systeem dat op deze wijze is gegroeid heeft een aantal beperkingen die toepassing buiten de kring van computerprogrammeurs uitsluit:

- Het kennis-niveau van de gebruiker m.b.t. de computer en het bedrijfssysteem moet minstens gelijk zijn aan dat van een programmeur, omdat bij de aanpassingen uitgegaan is van programma's waarvoor dat al vereist wordt.

- De te verwerken teksten moeten kunnen worden gerepresenteerd door dezelfde (meestal beperkte) tekenvoorraad die ook geschikt is voor de representatie van programmateksten.
- De lengte van de te verwerken teksten moet klein zijn (in de orde van grootte van een programma). Er staat doorgaans een progressief toenemende straf op het verwerken van langere teksten in termen van rekentijd, opslagkosten en turn-around tijd.
- Een bijkomend nadeel is dat bij vervanging van de computer installatie het tekstverwerkend systeem ophoudt te bestaan.

De wijzigingen die in een bedrijfssysteem moeten worden aangebracht om bovengenoemde beperkingen en nadelen op te heffen zijn zo ingrijpend, dat besloten is op het MC van de grond af een nieuw systeem te ontwikkelen. Dit systeem wordt een z.g.n. special purpose systeem, voor tekstverwerking. Op deze wijze hopen de ontwerpers de omvang van het project binnen de mogelijkheden van beschikbare mankracht en apparatuur te houden. Het systeem zal deel uitmaken van een bescheiden computer-netwerk, zodat ook combinatie van tekstverwerking met andere toepassingen (b.v. computer graphics, zie [42]) mogelijk wordt.

0.2 Een Systeem voor Ongeschoolde Gebruikers.

De aanmaak van gegevens voor computerverwerking gebeurt meestal off line door ponstypisten. Correcties worden aangebracht, hetzij door de ponstypist (off line), hetzij door de auteur zelf indien hij in staat is de computer met bijbehorende editor te gebruiken. Vaak is het geven van aanwijzingen meer werk dan het zelf aanbrengen. Off line corrigeren brengt ook met zich mee dat niet alleen de fout in de tekst, maar ook de omgeving van de fout moet worden overgetypt.

De moeilijkheden met het off line corrigeren vormen de belangrijkste aansporing om te trachten een methode te vinden om niet-programmeurs in staat te stellen direct met gegevens in de computer te werken. Uitgebreide analyses, alsmede pogingen om dit probleem aan te pakken zijn te vinden in [15], [18], [23], [27], [29]. Er zal een tekstinvoereenheid worden ontworpen die zowel on line als off line kan worden gebruikt, met minimaal verschil in bediening. Deze eenheid zal voorzien worden van een eenvoudige editor (eraser) en de mogelijkheid gegevens in hard copy vorm vast te leggen en weer in te lezen.

Een stap verder voert ons naar de gebruiker die zijn teksten een speciale bewerking wil laten ondergaan. In het systeem zullen speciale talen worden opgenomen die de gebruiker (b.v. een linguïst) de

mogelijkheid bieden zijn tekstbewerkingen met een minimale programmeerkennis zelf te specificeren. De eenvoudigste manier om dit te realiseren is door het ter beschikking stellen van een bibliotheek van tekstbewerkings procedures in de vorm van een zeer hoge programmeertaal. In een later stadium zullen editor en tekstbewerking per toepassing samengevoegd worden. Op deze wijze kunnen applicatiepakketten worden samengesteld. Voorbeelden hiervan zijn te vinden in [11], [20], [30].

De bruikbaarheid van het systeem wordt in hoge mate bepaald door het gemak waarmee dergelijke subsystemen gegenereerd kunnen worden. Het is duidelijk dat voor de gebruiker de kern van een tekstverwerkend systeem bestaat uit de editor en de edit-terminal. Op het MC heeft W. Wakker een studie gemaakt van een aantal bestaande editors ([27], [33], [34], [36], [37], [38]), de resultaten hiervan, neergelegd in [43], aangevuld met gegevens uit [33], [34] en [41] zijn verwerkt in hoofdstuk 3 waar een ontwerp voor een multi-user editor wordt gegeven.

0.3 Opslag van Teksten.

De bewaarcapaciteit van een tekstverwerkend systeem hoeft niet groot te zijn; voldoende voor de teksten onder behandeling. Gezien de grote variatie in aangeboden teksten moet de flexibiliteit t.a.v. data representatie in principe onbeperkt zijn.

Het systeem zal een uitbreidbare (dus onbeperkte) teken voorraad beheren.

Iedere aangeboden tekst wordt gerepresenteerd m.b.v. een eindige selectie uit de voorraad. Het kiezen van de juiste tekens en het produceren van een editor voor een gegeven tekenverzameling is specialistenwerk. Het resulterende tekstmanipulatiesysteem kan in principe weer toegankelijk zijn voor een gewone gebruiker. Voorbeelden van dergelijke reeds bestaande systemen zijn te vinden in [10] en [24].

Het beschikbaar stellen van een uitgebreide tekenvoorraad brengt met zich mee dat er aan in- en uitvoerkant veel conversieroutines moeten bestaan. Indien voor iedere tekstbewerking (w.o. opslag) de meest geschikte tekenvoorraad wordt gekozen, moeten ook intern file-conversies mogelijk zijn. Met een subsysteem voor tekstcompactificatie en decompactificatie zoals bv. in [9], [13], [35], [39] en [40] kan het systeem als front end voor een data bank functioneren. Verder moet een tekst die geanalyseerd wordt kunnen worden voorzien van een netwerk van verwijzingen hetgeen betekent dat tekstfiles gemengd moeten kunnen worden met files die besturingscodes bevatten.

0.4 Invoer en Uitvoer van Teksten.

Een belangrijke faciliteit van een tekstverwerkend systeem is de uitvoer naar een grote verscheidenheid van apparaten (terminals, printers, zetmachines, databanken en tekenmachines). In het licht van de beschouwing in de vorige paragraaf is uitvoer een bijzonder geval van tekstrepresentatie. Ieder apparaat is in het systeem gerepresenteerd door een tekenvoorraad.

Voor de ontwikkeling van programmatuur is het gewenst over een afdruk eenheid te beschikken waarop het totale bestand aan tekens kan worden afgedrukt, zodat iedere tekstfile in welke representatie dan ook kan worden zichtbaar gemaakt. Voor een dergelijke universele printer moet een beroep gedaan worden op een tekenmachine van het grafische systeem [42]. Vanaf deze machine vindt tevens de tekenvoorraad beheer plaats.

Het in- en uitvoer systeem voor tekstfiles moet tevens als interface dienen voor tekstbewerkingen die worden uitbesteed via het netwerk.

Door het gemak waarmee vreemde teksten kunnen worden geadopteerd, kan elders ontwikkelde programmatuur (w.o. eigen programmas in het verleden op andere installaties ontwikkeld, zie [1], ... , [8]) eenvoudig operationeel worden gemaakt.

Een prototype van een combinatie van editor en geavanceerde uitvoer is een typesetting systeem (zie [12], [16], [19], [25]). Combinatie van een typesetting systeem met een grafisch systeem zoals in [14], [22], [28], [31] en [32] stelt ons in staat geavanceerde typografie uit te voeren voor specialistisch en/of moeilijk zetwerk zoals kranten opmaak, mathematisch zetwerk of analyse van chinese tekst.

1. Doel van het tekstverwerkend systeem.

1.1 Exploitatie en Onderzoek.

Het tekstverwerkend systeem zal in de eerste plaats gebruikt worden voor de ondersteuning van activiteiten op het Mathematisch Centrum waarbij tekstverwerking een rol speelt. Voorbeelden zijn o.a. het vervaardigen van eigen boeken en rapporten, het ontwikkelen en documenteren van programma's, en het analyseren van teksten in een natuurlijke of kunstmatige taal.

Tijdens het ontwerpen en implementeren van het systeem zal eigen onderzoek noodzakelijk zijn. Reeds bekende hulpmiddelen voor tekstverwerking kunnen verbeterd worden. Het systeem zal toepassingen introduceren die, voorzover dat kon worden nagegaan, nog nergens mogelijk zijn.

In de tweede plaats zal het tekstverwerkend systeem gebruikt worden bij eigen onderzoek. De gedachten gaan uit naar ontwikkeling van speciale talen voor stringmanipulatie, hulpmiddelen om het verband tussen vorm en betekenis te beschrijven en onderzoek naar formele eigenschappen van talen.

In de volgende paragrafen van dit hoofdstuk zullen de faciliteiten uit het systeem gegroepeerd worden rondom belangrijke toepassings- of onderzoeksgebieden. De projektgroep tekstverwerking zal, in een vroeger of later stadium, bij elk van de ter sprake komende onderwerpen actief betrokken zijn, door (her-)opening van eigen onderzoek.

1.2 Tekstverwerking voor Publikaties.

De publikatiedienst van het Mathematisch Centrum zal een van de belangrijkste gebruikers van het systeem zijn. De programmatuur die voor dat doel ontwikkeld moet worden bestaat uit een aantal modules die nu achtereenvolgend besproken zullen worden.

1.2.1 Teksteditor, speciaal geschikt voor het intypen van manuscripten.

Dit werk moet uitgevoerd kunnen worden door de typistes van de typekamer. Op grond hiervan moet de editor aan de volgende eisen voldoen:

- Ze moet eenvoudig te bedienen zijn.

- De typesnelheid mag niet afnemen als gevolg van het gebruik van de editor (responstijd!).
- Het inbrengen van layout-aanwijzingen en het selekteren van nieuwe symbolensets moet eveneens snel en eenvoudig uitvoerbaar zijn.
- De typiste moet zich kunnen overtuigen van de juistheid van de gekozen symbolen en layout-instructies. Dit houdt in dat de editor de diverse symbolen en layout-procedures moet kunnen oproepen.
- Vervolgens moeten korrekties op de tekst en aanwijzingen eenvoudig in te brengen zijn.

In verband met andere toepassingen moet de teksteditor over een uitgebreid instructierepertoire beschikken. Voor speciale gevallen moeten nieuwe instructies kunnen worden ingebracht of worden samengesteld uit bestaande instructies. Tevens moeten (minimale) subsets uit het repertoire kunnen worden afgezonderd. Dit laatste moet gekombineerd kunnen worden met gestandaardiseerde layouts, zodat fout-detectie tijdens het intypen mogelijk is.

1.2.2 Proefdrukprogramma.

Het systeem moet op goedkope en snelle wijze een proefdruk kunnen leveren, die een natuurgetrouw beeld geeft van het eindresultaat. De fouten opgespoord aan de hand van de proefdruk moeten rechtstreeks onder de teksteditor kunnen worden verbeterd, d.w.z. zonder voorkodering door een kenner van het instructie-repertoire van de editor.

1.2.3 Layout-programma's.

De layout-programma's moeten in staat zijn een ruwe tekst voorzien van aanwijzingen om te werken tot een volledige publikatie. De vervaardiging van de drukvorm moet mogelijk zijn zonder correctie en, in een later stadium, zonder montage. Dit betekent dat alle tekstbewerkingen zoals: pagineren, alinea's aanbrengen, uitlijnen, afbreken, tabellen en voetnoten maken, het genereren van indextabellen en kruistabellen door het systeem verzorgd worden. Later moet de mogelijkheid om teksten en illustraties automatisch te mengen montage overbodig maken.

1.2.4 Korrektieprogramma's.

Er zullen speciale programma's ontwikkeld worden om fouten op te sporen zoals: typefouten, fouten tegen de layout en fouten tegen de systematiek van gestandaardiseerde documenten (b.v. mathematische bewijzen).

Het vastleggen van layout-voorschriften die onder meer layout-programma's besturen en het opsporen van afwijkingen mogelijk maken, vereist ontwikkeling van speciaal voor dat doel geschikte talen. Deze talen zouden ook een bijdrage moeten leveren aan het streven naar normalisatie van bijvoorbeeld mathematische notaties.

1.2.5 Speciale teksten.

Het systeem zal ook speciale teksten moeten kunnen behandelen, die gepubliceerd moeten worden zoals:

- computerprogramma's;
- mathematische-, chemische- en biologische formules;
- braille.

Voor elk van deze gevallen moet een bibliotheek van hulpprogramma's aanwezig zijn.

1.3 Tekstverwerking en Computergraphics.

Er loopt geen duidelijke scheidslijn tussen tekstverwerking en grafische toepassingen. Enerzijds kan een stuk tekst beschouwd worden als een bijzonder geval van een tekening, anderzijds kent bijna ieder grafisch systeem een toestand waarin tekstverwerking mogelijk is.

Op het Mathematisch Centrum wordt naast een tekstverwerkend systeem ook een grafisch systeem opgezet. Beide systemen zullen tijdens de ontwikkeling en tijdens de exploitatie van elkaars diensten gebruik maken.

De grafische groep zal in de ontwikkelingsfase gebruik maken van de faciliteiten die het tekstverwerkende systeem biedt voor programma-ontwikkeling en -dokumentatie (editor, uitlijner, PDP-dokumentatiesysteem enz). Voor de behandeling van blokken tekst, die onderdeel vormen van een tekening zal het grafisch systeem een beroep doen op de tekstverwerking. Voor het grafische systeem kan op die wijze een blok tekst beschouwd worden als een zwarte doos waarvan alleen enige uitwendige eigenschappen zoals afmeting en positie bekend

hoeven te zijn. De besturing van tekstproducerende randapparatuur is in principe ondergebracht bij de tekstverwerking (regeldrukkers, terminals).

Het tekstverwerkende systeem kan alle tekenwerk aan het grafische systeem delegeren. In feite beschouwen beide systemen elkaar als gebruiker. Alle apparatuur die tegelijk geschikt is voor tekenwerk en tekstuitvoer wordt ondergebracht in het grafische systeem (b.v. HRD-1 laser display/plotter en de DIGISET foto typesetter).

Analoog aan de manier waarop het grafische systeem met stukken tekst werkt, kan het tekstverwerkende systeem met tekeningen werken. Beide systemen, werkzaam in dezelfde computerconfiguratie (mini computer netwerk) moeten met korte responstijd opdrachten voor elkaar kunnen uitvoeren. Voor de gebruiker ziet het eruit alsof beide interactieve systemen over een combinatie van tekstverwerkende en grafische hulpmiddelen beschikken. In het bijzonder kan het grafische systeem behulpzaam zijn bij het ontwerpen en definiëren van symbolen (alfabetten) en bij het monteren.

1.4 Onderzoek naar Visuele Hulpmiddelen om betekenis te benadrukken.

Het is een goede gewoonte om programma's in een proceduregerichte taal zo te schrijven dat de blokstructuur en de nesting van de statements door de layout zichtbaar worden. Dit is een voorbeeld van een schrijfwijze waarbij de vorm van de tekst de logische opbouw van de inhoud verduidelijkt. Iets dergelijks is mogelijk voor zgn. niet-lineaire teksten, dat zijn teksten die niet noodzakelijk sequentieel gelezen hoeven te worden. Voor publikatiedoeleinden zijn er op het Mathematisch Centrum twee programma's beschikbaar die ALGOL60-programma's volgens dit principe behandelen. Het is niet ondenkbaar dat soortgelijke programma's kunnen worden ontwikkeld voor het behandelen van teksten voor syntactische definities, programma-dokumentatie (zie ...), mathematische theorieën (stelling, bewijs, voorbeelden enz.) en literatuuroverzichten. Een zeer interessant uitgangspunt daarbij is de mogelijkheid te scheppen om teksten een syntactische structuur af te dwingen op grond van een zo gering mogelijke redundantie van de erin vervatte gegevens.

Deze aspecten van een tekstverwerkend systeem vormen ook de basis voor de interactie met databanken. Daarbij is het van belang dat uittreksels, associatieve bestanden e.d. aan de gebruiker op een zo overzichtelijk mogelijke wijze kunnen worden gepresenteerd. Ook de opbouw van zo'n bestand bestaat voor een groot gedeelte uit tekstbewerkingen.

Het ligt in de bedoeling dat de grafische groep een databank voor tekeningen gaat aanleggen. Vervolgens zal een poging gedaan worden tekst en tekeningen in een bestand onder te brengen. Bij dat project zal er worden samengewerkt door beide onderzoeksgroepen.

1.5 Tekstanalyse.

het tekstverwerkend systeem zal, geruggesteund door nieuwe ontwikkelingen op het gebied van programmeertalen, het onderzoek naar formele eigenschappen van talen (natuurlijke of kunstmatige) vergemakkelijken. Daarbij zal, evenals dat in het verleden het geval is geweest, contact gezocht worden met linguïsten.

Op het gebied van probleemgerichte talen zullen talen ontwikkeld worden voor stringmanipulatie en (later) talen voor het vastleggen van associaties en soortgelijke niet-lineaire betekenisstructuren. In deze talen moeten met name zoek- en sorteerprogramma's kunnen worden beschreven. Deze zijn essentieel voor vergelijkend taalonderzoek.

2. Gebruik van de PDP8/I en PDP8/E Computers.

Het tekstverwerkend systeem en het grafisch systeem zullen samen worden geïmplementeerd in een computernetwerk bestaande uit een PDP11/45, een PDP8/I en een PDP8/E computer. Dit netwerk is tevens verbonden met de CDC-Cyber computer van de Stichting Academisch Reken-centrum Amsterdam.

Het tekstverwerkend systeem zal grotendeels worden ondergebracht in de PDP8/I en de PDP8/E; het grafisch systeem zal hoofdzakelijk worden ondergebracht in de PDP11/45. In enkele gevallen zal voor tekstverwerking een beroep worden gedaan op de PDP11 met zijn randapparatuur, omgekeerd zal het grafisch systeem vooral voor I/O gebruik willen maken van de aan de PDP8 gekoppelde randapparatuur. Beide systemen zullen taken aan de CDC-Cyber delegeren.

In dit hoofdstuk zal eerst worden aangegeven welke hardware bij de PDP8/I en PDP8/E beschikbaar is, daarna zullen enkele eisen worden geformuleerd die aan de voor deze machines te ontwikkelen software moeten worden gesteld. Hierbij hebben we ons niet strikt beperkt tot datgene wat nodig is voor tekstverwerking, veeleer is gelet op de vereisten voor een goed samengaan van beide systemen in het netwerk. Tenslotte is op een aantal plaatsen vermeld hoe bij de ontwikkeling van deze software kan worden voortgebouwd op werk dat reeds eerder op het MC werd gedaan.

2.1. Hardware.

2.1.1. Thans aanwezige en binnenkort te verwachten apparatuur.

- 1 32k PDP8/I;
- 1 32k PDP8/E;
- 2 DECTape units;
- 1 fixed-head disk;
- 2 disk cartridge units;
- 1 plotter;
- 2 bandponcers;
- 2 bandlezers;
- 2 regeldrukkers;

- 1 brailledrukker;
- 1 teletype;
- 1 storage display + joystick cursor;
- 3 D/A converters;
- 1 koppeling PDP8/I <-> PDP8/E;
- 1 koppeling PDP8/I <-> PDP11/45;
- 1 koppeling PDP8/E <-> PDP11/45.

2.1.2. Terminals.

De volgende terminals zullen aan het systeem worden gekoppeld:

- Hardcopy type:
 - Olivetti Te318 (110 Bd);
 - Datacare HCT 302 (110-300 Bd);
 - Triformation LED-120 brailleterminal (1200 Bd);
 - draagbare terminal.
- Display type:
 - Hewlett-Packard HP2640A (110-2400 Bd);
 - storage display (zie 2.1.1.) in combinatie met toetsenbord van een terminal of teletype.

Deze diversiteit stelt ons in staat ervaring op te doen met verschillende typen terminals. Op de PDP8 computer kan voor elk van deze terminals een driver worden geïmplementeerd, die een hoogwaardige real-time service biedt aan de gebruiker. Evaluatie van de opgedane ervaring kan leiden tot specificatie van eisen aan de terminal hardware en/of de driver software.

2.2 Programmatuur.

Teneinde de hieronder te beschrijven plannen op het gebied van software te verduidelijken, volgt eerst een schetsmatige indeling van de beschikbare geheugenkapaciteit.

- 32K PDP8/I: 8K bedrijfssysteem;
12K-8K buffers;
12K-16K OS/8.
- 32K PDP8/E: 4K bedrijfssysteem;
4K I/O buffers;
4K multi-user editor;
12K tekstbuffers multi-user editor;
8K toekomstige uitbreidingen.
- 256K RF08: 128K bedrijfssysteem + 128K buffers.
- 1600K RK08: 600K multi-user editor + 600K buffers;
400K OS/8 filesysteem.
- 1600K RK08: verwisselbare (permanent-)file-opslag.

2.2.1. Bedrijfssysteem.

De beschikbare randapparatuur is zoveel mogelijk gekoppeld aan de PDP8/I (de RK08-schijven konden niet aan de PDP8/I worden gekoppeld). De PDP8/I zal dan ook in de eerste plaats worden gebruikt voor het transport van data van en naar randapparatuur ten behoeve van programma's die in de PDP8/E of PDP11 worden uitgevoerd. Voorzover de processorkapaciteit dit toelaat zal (op achtergrond) de PDP8/I ook gebruikt kunnen worden voor het ontwikkelen en testen van programma's.

Uitgaande van bovengeschetst gebruik en configuratie kan het bedrijfssysteem voor de PDP8/I als volgt worden opgezet.

Het systeem kent drie besturingstoestanden:

- de interrupttoestand: Er is een interrupt geweest, de machine is doof. De interrupt wordt afgehandeld. Een groot deel van de besturing van randapparatuur gebeurt in deze toestand.
- De taaktoestand: De machine is horend. In deze toestand worden een aantal simultaan verlopende processen (taken) uitgevoerd. Ook het scheduleren van deze taken en het verzorgen van de communicatie tussen deze taken geschiedt in de taaktoestand. Zodra er een interrupt komt gaat de machine over in de interrupttoestand. Kan er

tengevolge van het afhandelen van de interrupt worden verder gerekend aan een taak met hogere prioriteit dan die van de taak waaraan werd gerekend voor de interrupt kwam, dan wordt de laatste onderbroken en wordt verder gerekend aan de taak van hoogste prioriteit.

- De achtergrond-toestand: de machine is horend, aan geen enkele taak kan worden verder gerekend. Zodra er een interrupt komt gaat de machine over in de interrupttoestand. Zodra weer aan een taak kan worden verder gerekend gaat de machine over in de taaktoestand.

Buiten de interrupttoestand bestaan slechts voor het reageren op de interrupts van enkele apparaten real-time verplichtingen, zoals voor de DECTape (maximum responstijd 400 usec.) en de snelle bandlezer (maximum responstijd 1 msec.). Het is wenselijk dat de interrupts van beeldschermen, de verbinding met de PDP8/E en de verbinding met de PDP11 snel worden afgehandeld.

In de taaktoestand bestaan geen real-time verplichtingen, hoewel elk toetsenbord in principe eens per 0.1 sec bediend moet kunnen worden. De capaciteit van het kerngeheugen laat niet toe dat de kode voor alle tot het bedrijfssysteem behorende taken zich permanent in het kerngeheugen bevindt. Voor de meeste taken is dit geen vereiste. Taken moeten dynamisch van (en naar) het achtergrond-geheugen worden geswapt. Geheugentoe wijzing voor taken en buffers dient dynamisch te geschieden, taken moeten dus relocateerbaar geschreven zijn.

Het is denkbaar dat verschillende taken tegelijkertijd van eenzelfde resource (geheugenruimte, I/O apparaat, derde taak) gebruik willen maken.

Om ongewenste interferenties bij het gebruik van bepaalde resources te voorkomen, is een claim-mechanisme vereist dat resources aan taken kan toewijzen. Deadlocks moeten worden uitgesloten.

De bij het bedrijfssysteem behorende scheduler bepaalt in welke volgorde processen worden uitgevoerd. Het is belangrijk onderscheid te maken tussen urgente en minder urgente processen, m.a.w. aan elk proces moet een zekere prioriteit worden toegekend.

Achtergrond-programma's worden slechts uitgevoerd, als er naast het afhandelen van interrupts en taken nog voldoende processorkapaciteit beschikbaar is en aan het achtergrond-programma voldoende resources zijn toegewezen. Het lijkt aantrekkelijk OS/8 - een single-user systeem - te implementeren. Dit systeem kan worden gebruikt voor software ontwikkeling. Het taakniveau en het interruptniveau moeten beschermd kunnen worden tegen "foutieve" achtergrond-programma's (protektie): dat wil zeggen, dat het achtergrond-programma geen toegang

heeft tot geheugen of randapparatuur die niet aan het achtergrondprogramma zijn toegewezen. Protèktie wordt in de PDP8 gerealiseerd door delen van het onder protèktie draaiende programma niet rechtstreeks uit te voeren. Deze delen worden aangeboden aan een simulator, die ze uitvoert indien dit mogelijk is in de toegewezen resources. Deze simulator is een taak in het bedrijfssysteem, die in het geheugen dat aan een achtergrond programma is toegewezen een PDP8 machine simuleert.

De PDP8/E zal primair worden gebruikt voor het editen van teksten. Er zal een programma worden geschreven (multi-user editor) dat meerdere gebruikers simultaan in staat stelt elk hun eigen tekstfile te editen. Voor het scheduleren van verschillende gebruikers en het verzorgen van I/O doet de multi-user editor een beroep op een in de PDP8/E onder te brengen bedrijfssysteem, dat daarnaast nog tot taak heeft het verzorgen van datatransporten van en naar de RK08-schijven, en het besturen van het Extended Arithmetic Element (o.a. ten behoeve van de plotter). Het bedrijfssysteem op de PDP8/E zal een subsysteem zijn van dat op de PDP8/I, aangevuld met enkele taken (b.v. een taak voor het verrichten van floating point berekeningen).

Resumerend kunnen we zeggen, dat het bedrijfssysteem voor de PDP8/I de volgende functies moet kunnen vervullen:

- Afhandelen van interrupts met een maximum respons-tijd van ca. 350 usec..
- Afhandelen van interrupts in een gemiddelde tijdsduur van ca. 250 usec. per interrupt (bij interruptfrequentie van 1000 interrupts per seconde, 25% van de processorkapaciteit).
- Dynamisch swappen van taken van en naar het achtergrondgeheugen.
- Dynamische geheugentoewijzing.
- Verplaatsen van taken.
- Behandelen van zgn. claims met uitsluiting van deadlock situaties.
- Besturen van taken met verschillende prioriteit.
- Het uitvoeren van achtergrond-programma's onder bescherming van de overige taken.

En aantal bestaande bedrijfssystemen op PDP8 computers is door ons bestudeerd. Uit onderstaande opsomming blijkt, dat geen van deze systemen aan alle eisen voldoet.

- OS/8 Single-user bedrijfssysteem, geschikt om als bedrijfssysteem voor achtergrond-programma's te worden geïmplementeerd. Echter, OS/8 is niet interrupteerbaar.
- RTS8 Systeem met interrupt-, taak- en achtergrond-toestand. Het aantal taken is statisch bepaald en de taken staan in principe permanent in het kerngeheugen; ze behoeven dus ook niet relocateerbaar te zijn. Geen claim-mechanisme. Protektie is matig geïmplementeerd. Op het Mathematisch Centrum zijn wijzigingen ontworpen, waardoor dynamisch in- en uitswappen van taken in principe mogelijk wordt.
- MULTI-8 Systeem met interrupt-, taak- en achtergrond-toestand. In de achtergrond is timesharing mogelijk. Het claim-mechanisme kent geen deadlockpreventie. De taakscheduler houdt geen rekening met taakprioriteiten. Protektie is slecht geïmplementeerd.
- TRAC Systeem met interrupt-, taak- en achtergrond-toestand. Dit systeem werd op het MC ontwikkeld. Dynamisch in- en uitswappen van taken is niet mogelijk. Er is geen claim-mechanisme.

2.2.2. Filestysteem.

Op het beschikbare schijfengeheugen zullen een aantal filesystemen worden geïmplementeerd. De schijfruimte die door het bedrijfssysteem voor buffers wordt gebruikt, wordt toegewezen door een eenvoudig dynamic storage allocation system met garbagecollector. Op het voor OS/8 gereserveerde deel van de schijfruimte, zal het OS/8 filestysteem worden geïmplementeerd.

Meer aandacht dient te worden besteed aan het filestysteem dat de multi-user editor zal gebruiken. Hieraan worden de volgende eisen gesteld:

- Files moeten in principe random accessible zijn.
- Toevoegen en weglaten van gedeelten van files moet snel kunnen geschieden.
- Editen op verschillende files moet simultaan mogelijk zijn.
- Garbagecollection moet tijdens het editen kunnen worden gedaan.

Een filesysteem dat aan deze eisen voldoet werd door ons ontwikkeld en in ALGOL60 en ALEPH geïmplementeerd.

2.2.3. Editor.

Om te komen tot een optimaal ontwerp van het kommandorepertoire van de multi-user editor, is door ons enige literatuur en een groot aantal edit- en uitlijnprogramma's bestudeerd (zie ook MC-memorandum over editors door W. Wakker). Veel ervaring werd opgedaan bij het gebruik van de door DEC bij OS/8 geleverde editor. Hierin is een groot aantal wijzigingen aangebracht om performance en kommandorepertoire te verbeteren.

2.3. Gebruikte programmeertalen.

Aan de efficiëntie van het bedrijfssysteem en de daarin opgenomen taken, worden hoge eisen gesteld. Het ligt dan ook voor de hand om deze programmatuur in PDP8 assemblercode te schrijven.

Ook de multi-user editor zal in assemblercode geschreven worden om het programma zo efficiënt mogelijk te maken. Weliswaar komt dat de overdraagbaarheid van het programma naar andere machines en de leesbaarheid niet ten goede, maar delen van de multi-user editor zullen zeer nauw met het bedrijfssysteem zijn verweven, zodat overbrengen naar een andere machine niet in aanmerking komt.

Daarentegen zullen de meeste tekstverwerkende programma's in een hogere programmeertaal worden geschreven; immers dit bevordert overdraagbaarheid naar andere machines, de leesbaarheid van programma's en het vereenvoudigt het ontwikkelen van programma's. In verband hiermee is het noodzakelijk implementaties van ALGOL-achtige talen, ALEPH en de nog te ontwikkelen stringmanipulatietalen op het netwerk operationeel te maken. Een beperkte ALGOL60 implementatie werd reeds op het Mathematisch Centrum ontwikkeld.

3. Structuur.

Het tekstverwerkend systeem is opgebouwd uit een aantal lagen. Op het onderste (eerste) nivo bevindt zich het bedrijfssysteem. Het bedrijfssysteem is niet speciaal gericht op tekstmanipulatie. Alleen die delen, die rechtstreeks met tekstverwerking verband houden zullen worden besproken. Op het tweede nivo bevinden zich programma's voor het aanmaken en korrigeren van teksten. Een tekst is een rij "beschreven posities"; een beschreven positie bestaat uit een symbool en een positie. De symbolen van een tekst zijn veelal gegroepeerd in pagina's, onderverdeeld in regels, onderverdeeld in symboolplaatsen. In een dergelijke tekst kan men de positie aangeven door 3 getallen: paginanummer, regelnummer en symboolplaatsnummer. Een dergelijke tekst heet lineair als voldaan is aan onderstaande voorwaarde:

Laten B_m en B_n beschreven posities zijn uit de tekst met $n > m$, met posities (P_m, R_m, K_m) en (P_n, R_n, K_n) . de tekst heet lineair als geldt:

$P_n > P_m$
of $P_n = P_m$ en $R_n > R_m$
of $P_n = P_m$ en $R_n = R_m$ en $K_n > K_m$.

Teksten waarvan de positie niet op deze wijze kan worden aangegeven, of waarvan de posities niet aan bovenstaande voorwaarde voldoen, heten niet-lineair.

Symbolen zijn verdeeld in alfabetten, elk alfabet omvat 95 symbolen. Teksten worden opgeslagen als een rij van 8-bits symbolen:

- 128 command symbolen;
- 32 controle symbolen;
- 1 delete symbool;
- 95 alfabet symbolen.

Command symbolen worden gebruikt voor het besturen van de editors of van applicatieprogramma's (b.v. het kiezen van een alfabet). Control symbolen hebben veelal een standaard functie (b.v. carriagere-turn, linefeed, e.d.). Het delete symbool wordt gebruikt voor correctie. De alfabet symbolen representeren de symbolen van de tekst.

Voor de aanmaak van teksten beschikt het systeem over editors: een multi-user editor voor teksten met een lineaire structuur en single-user editors voor niet-lineaire teksten.

De werkruimte nodig tijdens tekstbewerking en de ruimte nodig voor het opslaan van teksten is georganiseerd in de vorm van files.

Editors en filesysteem samen vormen het tweede nivo.

Het derde nivo bevat alle applicatieprogramma's. Voorzover nodig kan in het totale pakket applicaties weer een gelaagde structuur aangebracht worden.

De twee onderste lagen zijn permanent in het systeem aanwezig. Een terminal gebruiker kommuniqueert met het onderste nivo uitsluitend om toegang te krijgen tot een van de editors of om een applicatieprogramma te activeren en te besturen.

Vanaf het tweede nivo kan er onderscheid gemaakt worden tussen lineair en niet-lineair werk. Bij de implementatie zal eerst het werken met lineaire teksten en vervolgens het werken met niet-lineaire teksten gerealiseerd worden.

De interface met het grafische systeem bevindt zich gedeeltelijk op het onderste nivo en voor een gedeelte binnen het filesysteem uit de tweede laag. Zolang uitsluitend gewerkt wordt met lineaire teksten zal vanuit het tekstverwerkende systeem geen beroep gedaan worden op het grafische systeem. Omgekeerd zal het grafische systeem wel gebruik maken van de faciliteiten van het tekstverwerkende systeem. Pas in de tweede fase zal het grafische systeem diensten verlenen aan de tekstverwerking, zodat de interface dan in beide richtingen moet kunnen werken.

In de volgende paragrafen zal aan de hand van de hierboven gegeven indeling nader op de diverse modulen worden ingegaan.

3.1 Het onderste nivo.

De hogere nivo's bestaan uit families van taken. Het onderste nivo zorgt voor de besturing van alle taken.

Een familie van taken voert een bepaald proces uit. Iedere taak doet een beroep op resources van het systeem, waarbij inbegrepen de diensten van andere taken. De toekenning van resources en de dienstverlening verloopt volgens bepaalde regels. Een moduul uit het onderste nivo, de zgn. scheduler, garandeert dat deze regels in acht worden genomen.

De scheduler realiseert een prioriteitenmechanisme met behulp waarvan de real-time verplichtingen van de diverse randapparaten kunnen worden nagekomen. De scheduler beschikt over een aantal routines om de door taken aangevraagde resources toe te wijzen. De communicatie tussen taken onderling loopt via boodschappen. Het boodschappenmechanisme bevindt zich eveneens in de scheduler. Voor de communicatie met het systeem beschikken de taken over virtuele instructies die door de scheduler worden geïnterpreteerd.

Het is duidelijk dat op het onderste nivo geen instructies bestaan die specifiek zijn voor tekstverwerking.

3.2 Multi-User Editor en Filesystem.

De multi-user editor (MUE) is een familie van taken, die een aantal terminalgebruikers in staat stelt, simultaan lineaire teksten aan te maken of te korrigeren. De MUE vormt de kern van het tekstverwerkend systeem en moet geschikt zijn voor het editen van een grote verscheidenheid van teksten. In het tekstverwerkend systeem wordt met meerdere alfabetten gewerkt. Wisseling van alfabet kan met behulp van command symbolen worden aangegeven. Niet alle command symbolen worden door de editor herkend, sommige zijn alleen voor applicatieprogramma's herkenbaar.

De kwaliteit van de editor wordt bepaald door de syntax en semantiek van de kommandotaal en de responstijd (tijd nodig voor het uitvoeren van de kommando's).

Het kommandorepertoire is opgebouwd rondom een set basisinstructies. Deze basisinstructies moeten een ongetrainde gebruiker in staat stellen, snel en foutloos edit-kommando's te geven. Met behulp van deze basisinstructies kan in principe elk edit-werk gedaan worden. Dit betekent:

- Iedere substring van een tekst kan vervangen worden door iedere andere substring.
- Het resultaat kan zichtbaar gemaakt worden na ieder kommando.

Het is van belang de basis instructies zo te kiezen dat het effect op de tekst ook voor een ongetrainde gebruiker onmiddellijk duidelijk is. De volgende drie basisinstructies zijn gekozen:

- Selektie van een substring. Een substring wordt geselecteerd door zijn beginpositie en zijn eindpositie op te geven.
- Vervanging van een geselecteerde substring door een andere.
- Het zichtbaar maken van een substring met zijn omgeving.

Zowel de geselecteerde string als de ter vervanging aangeboden string kunnen leeg zijn. Posities in de tekst worden aangeduid door specificatie van pagina-, regel-, en symboolplaatsnummer. Het basisrepertoire wordt uitgebreid tot een compleet editrepertoire door:

- Implementatie van een stringsearch, waardoor selektie van substrings eenvoudiger wordt.

- Het aanwijzen van de ter vervanging aangeboden string door selectie; dit maakt mengen van tekstgedeelten mogelijk.
- Invoering van relatieve positionering. Dit vereenvoudigt in veel gevallen de kommando's.
- Gebruik te maken van per toepassing te specificeren standaardwaarden.
- Implementatie van een mechanisme om nieuwe kommando's te vormen uit reeds bestaande; Dit mechanisme wordt in de vorm van een kommandotaal aan de gebruiker gegeven. Deze taal bevat control statements voor opeenvolging van kommando's, loops en conditionele statements.

De responstijd wordt minimaal gemaakt door zoveel mogelijk regelwijs van de terminal in te lezen. Zolang de regel niet is afgesloten, kan deze zonder tussenkomst van de editor gewijzigd en getoond worden. Toepassing van wisselbuffers zorgt ervoor, dat een nieuwe regel kan worden ingetypt, terwijl de vorige verwerkt wordt.

De MUE werkt met teksten die in files zijn opgeslagen. Een goede editor staat toe, dat de gebruiker de wijzigingen in een tekst in een willekeurige (= niet noodzakelijk sequentiele) volgorde aanbrengt. Er is dus een zekere mate van random access tot de file nodig.

Verder moet het toevoegen of weglaten van stukken tekst mogelijk zijn zonder grote delen van de file te kopiëren.

Een file bestaat uit een aantal genummerde blokken (logische bloknummering). Het eerste blok bevat informatie over de manier waarop de pagina's over de blokken verdeeld zijn. De editor kan een stuk identificeren met filenaam en logisch bloknummer. Het filesysteem beheert een kruistabel die voor elke file het verband geeft tussen fysische en logische bloknummers.

Random access op bloknivo is voor de editor voldoende. Hij kan dan een willekeurig blok in zijn prive werkruimte plaatsen. Deze werkruimte is random toegankelijk. Blokmanipulaties worden door het filesysteem voor de editor op verzoek uitgevoerd.

Vanuit de editor kunnen subfiles ter verwerking aangeboden worden aan applicatieprogramma's. De applicatieprogramma's kunnen de subfiles na verwerking, al dan niet gewijzigd, teruggeven. Dit schept zeer ruime mogelijkheden tot interactieve tekstverwerking (interactief uitlijnen, interactief typefouten opsporen, enz.).

3.3 Niet-Lineaire Teksten.

In het pakket applicatieprogramma's zullen programma's zijn opgenomen die, uitgaande van een lineaire tekst, een niet-lineaire tekst produceren.

In het algemeen zal een niet-lineaire tekst op beperkte en regelmatige wijze afwijken van de lineaire structuur.

De variatie in de afwijkingen is evenwel zo groot, dat het niet mogelijk is om een algemene karakterisering van de structuur van deze teksten te geven. Als gevolg hiervan bestaat er geen editor die geschikt is voor elke niet-lineaire tekst. Er zal een kollekte editprogramma's worden ontworpen waarin de meest gangbare niet-lineaire patronen eenvoudig kunnen worden aangegeven.

De niet-lineaire structuur komt tot uitdrukking in de layout van de tekst. Het manipuleren met niet-lineaire teksten kan daarom alleen aan een beeldscherm gebeuren. Bij voorkeur zal men werken met een beeldscherm waarop het mogelijk is, een niet-lineair samenhangend gedeelte van de tekst te vervangen. Dit kan alleen aan een refresh-display dat, vanwege de niet-lineariteit van het beeld een grafisch display moet zijn.

De benodigde grafische displays moeten over een grote symboolkapaciteit beschikken, d.w.z. de voorraad tekens moet dynamisch kunnen worden vastgelegd, zonder dat de afdruksnelheid terugloopt. Dergelijke displays zijn nog niet beschikbaar. In de huidige displays zou het read-only memory met de symboolsjablonen vervangen moeten worden door een (groter) read-write memory.

Het werken met een display in refreshmode vergt een grote processor-kapaciteit. In het huidige tekstverwerkende systeem is hoogstens ruimte om experimenten te doen met zo'n display.

3.4 Interface met het Grafische Systeem.

Het tekstverwerkend systeem kent alleen teksten, het grafische systeem kent zowel teksten als tekeningen. Via de interface tussen beide systemen kunnen teksten worden uitgewisseld.

In het tekstverwerkend systeem is een tekening teruggebracht tot een naam en een stel layout-aanwijzingen die bewerkstelligen dat er ruimte voor de tekening tussen de tekst wordt gemaakt. Het tekstverwerkend systeem kan gegevens over een tekening via de interface opvragen.

De produktie van een dokument dat is samengesteld uit tekst en tekeningen zal in eerste instantie geschieden door de tekeningen in het grafische systeem, de teksten in het tekstverwerkend systeem aan te maken. De montage van het dokument vindt plaats aan een terminal

van het grafische systeem, waarbij de aangemaakte teksten en tekeningen worden opgeroepen. In dat stadium kan de layout nog worden gewijzigd.

Het tekstverwerkend systeem kommuniqueert met het grafische systeem op taaknivo: Het grafische systeem wordt beschouwd als een verzameling taken die geactiveerd kunnen worden. Vanuit het grafische systeem kunnen taken voor de behandeling van een blote tekst worden geactiveerd. Voor input en output van die taken moeten files ter beschikking staan.

Voor het bewerken van niet-lineaire teksten kan het tekstverwerkend systeem een beroep doen op het grafische systeem. Het grafische systeem wordt dan opgevat als een afdrukkapparaat of als een terminal. De eigenschappen van deze terminal die van belang zijn voor de tekstverwerking kunnen dynamisch worden vastgesteld.

3.5 Applikatieprogramma's.

De applicatieprogramma's zijn georganiseerd in bibliotheken. Per toepassing bestaat er een applicatiepakket van programma's, dat er voor het systeem uitzielt als een familie taken.

Niet alle applicatieprogramma's zijn in het systeem zelf geïmplementeerd, sommige draaien op andere configuraties. Van een dergelijke taak kent het systeem alleen de startkondities en de vorm van het eindresultaat in geval van een succesvolle uitvoering.

De resterende opmerkingen in deze paragraaf hebben betrekking op applicatieprogramma's die in het systeem worden uitgevoerd.

Zodra een taak uit de bibliotheek wordt opgeroepen, wordt zijn hele familie in het systeem gebracht, mits er voldoende resources zijn voor de hele familie. Kommunikatie tussen taken uit eenzelfde familie is nu eenvoudig, daar ze altijd tegelijk in het systeem aanwezig zijn. Kommunikatie tussen taken uit verschillende families moet verlopen via het filesysteem.

We zullen nu een aantal applicatiepakketten omschrijven, die onzes inziens onmisbaar zijn voor tekstverwerking. Het is duidelijk dat de kollektie programma's in principe onbeperkt kan groeien.

3.5.1 Layout-programma's.

Layout-programma's verdelen een tekst in fysische pagina's en positioneren de tekst op de pagina's.

Ieder layout-programma kent een machine-afhankelijk en een machine-onafhankelijk gedeelte (machine = afdrukeenheid). De machine-afhankelijkheid komt tot uitdrukking in paginahoogte, regel-

breedte en symbool-formaat (al of niet variabele korpsgrootte), etc.. Konversie van een tekst van een machine naar een andere vergt in het algemeen heruitvoering van het layout-programma (met de oorspronkelijke layout-aanwijzingen).

In het algemeen levert een layout-programma een voor de gebruiker niet optimaal resultaat af. Wat dit betreft kunnen we twee gevallen onderscheiden:

- Het algoritme wordt geheel bepaald door een aantal numerieke parameters (layout-aanwijzingen).
- In het algoritme is een of andere vorm van taalanalyse verwerkt. Voor dergelijke algoritmen is het essentieel dat twijfelgevallen gedetekteerd worden, en tijdens het rekenen worden voorgelegd aan de gebruiker. Deze programma's moeten daarom interaktief kunnen werken.

3.5.2 Afdruk- en konversie-programma's.

Bij iedere tekstproducerende machine hoort een afdrukprogramma. Zo'n programma stuurt een volledig uitgelijnde file naar een machine. Het programma kent de volledige besturingscode voor die machine. Het genereert een reeks I/O akties, die door een apparaat-taak worden afgehandeld. Tijdens het versturen vindt kodekonversie plaats en worden besturingscodes voor de machine ingebracht.

De tekstverwerkende programmatuur werkt zoveel mogelijk met een standaard interne representatie van de ingevoerde tekst. Om dit mogelijk te maken zijn er konversieprogramma's die in staat zijn aangeboden tekstfiles zonder informatieverlies te converteren naar deze standaard representatie.

3.5.3 Index- en sorteer-programma's.

Index- en sorteerprogramma's kunnen, evenals layout-programma's, verdeeld worden in twee categorieën, op grond van het feit dat het selektiemechanisme al of niet gebruik maakt van taalanalyse.

Deze programma's kunnen opgevat worden als gespecialiseerde editors, die al of niet interaktief werken. Het is de bedoeling dat deze programma's op eenvoudige wijze vanuit een editor kunnen worden aangeroepen.

Het kompleks van edit-, index- en sorteer-programma's vormt in principe de kern van een databank-beheersysteem.

3.5.4 Tekstanalyse.

Aktiviteiten op het gebied van tekstanalyse vertonen duidelijke verwantschap met werk op het gebied van kunstmatige intelligentie. We willen hier slechts enkele van de talloze mogelijkheden aanstippen:

- Analyse die leidt tot datareduktie (nuttig voor databanken).
- Analyse die leidt tot het detekteren van schrijffouten.
- Analyse van formele eigenschappen van talen (statistische, syntactische).

3.5.5 Beheer en ontwerp van symboolverzamelingen.

Het ligt in de bedoeling alle symboolverzamelingen (alfabetten), die op enigerlei wijze in het tekstverwerkende systeem gebruikt worden, op te slaan in een databank, tezamen met alle relevante gegevens, zoals:

- De apparatuur waarop de symbolen afgedrukt kunnen worden; met de bijbehorende kode.
- Het symboolformaat; van belang voor layout programma's.

Er moet tenminste een terminal zijn waarop het volledige beheer van de databank mogelijk is. Deze terminal zal ook gebruikt worden voor het ontwerpen en invoeren van nieuwe tekens. Daar deze tekens in eerste instantie uitsluitend als tekening herkenbaar zijn (pas bij invoering in de databank wordt een symboolidentifikatie toegevoegd), is deze terminal noodzakelijkerwijs een grafisch eindstation.

3.6 Fasering en Planning.

Het totale tekstverwerkende systeem zal in drie fasen gerealiseerd worden:

- fase 1: een systeem voor lineaire teksten;
- fase 2: uitbreiding naar een systeem voor niet-lineaire teksten;

- fase 3: selectie en samenstelling van programmapakketten in de vorm van stand-alone systemen voor speciale toepassingen.

Over geautomatiseerde tekstverwerking in de praktijk is nog zo weinig bekend, dat tijdens de ontwikkeling reeds proeven gedaan moeten worden om de bruikbaarheid van het systeem te meten. In verband daarmee zullen veel programma's in een vroeg stadium aan de gebruiker ter beschikking worden gesteld.

De realisatie van het project is verdeeld in vijf stappen, die ieder ongeveer een jaar vergen. De schatting van een jaar is niet alleen gebaseerd op de hoeveelheid programmeerwerk die verzet moet worden, doch houdt ook rekening met de hierboven genoemde toetsing aan de praktijk en de bijbehorende evaluatie.

Het vijfjarenplan ziet er als volgt uit:

- 1. Eerste versie van de multi-user editor en het filesysteem, proefdraaien met beide.
- 2. Operationeel maken van beschikbare applicatieprogrammatuur, definitieve versie van multi-user editor en filesysteem, aanzet tot nieuwe applicatieprogrammatuur.
- 3. Interface met het grafische systeem, ontwerp van een niet-lineaire editor.
- 4. Uitbreiding met applicatieprogrammatuur voor niet-lineaire teksten.
- 5. Opzet van databanken, complete applicatiepakketten .

4. Litteratuur.

- [1] P.J.W. ten Hagen en K.B. Thio,
Het tekenen van formules; beschrijving van de invoer,
MC Rapport NR 1, april 1967.
- [2] G.H.A. Kok,
Het automatisch omzetten van Nederlands in een fonetische notatie
MC Rapport MR 130, februari 1972.
- [3] H.C. Brandt Corstius,
Computer Typesetting,
Proc. Computer Typesetting Conference, London 1965, pp 183-185.
- [4] H.C. Brandt Corstius,
Exercises in Computational Linguistics,
MC Tract 30, Mathematisch Centrum, Amsterdam 1970.
- [5] J.C. van Vliet,
Beste Splits 1; een procedure voor het automatisch afbreken van
Nederlandse woorden,
MC Rapport NR 28, juni 1972, vertrouwelijk.
- [6] D. Grune,
Handleiding bij de Tekstcorrector,
MC LR 2.2, juli 1971.
- [7] D. Grune,
Handleiding bij de Tekstschaaf,
MC LR 2.5, april 1972.
- [8] J.A.Th.M. van Berkel et al,
Formal Properties of Newspaper Dutch,
MC Tract 12, Mathematisch Centrum, Amsterdam.
- [9] R.N. Ascher and G. Nagy,
A Means for Achieving a High Degree of Compaction on Scan-
Digitized Printed Text,
IEEE Transactions on Computers Vol. C-23, no 11, 1974.
- [10] P.G. Dowd et al,
STARE2 - A Graphical Hard-Copy System,
IEEE Transactions on Computers Vol. C-22, no 4, 1973.
- [11] P.G. Moore and R.P. Mann,
Cypher Text - An Extensible Composing and Typesetting Language.
FICC 1970, pp 555-561.
- [12] Alt, F.L. and Kirk J.Y.,
Computer Photo Composition of Technical Text,
CACM june 1973, Vol 16, no 6.
- [13] Suzuki R. et al,
On a Technique for Compact Generation of Symbols,
Inf. Proc. in Japan, Vol 11, 1971.
- [14] H. Kautto et al,
ICG as a new Tool for Newspaper editors,
- [15] D.W. Packard,
Can scholars publish their own books?
Scholarly Publishing, oct. 1973.
- [16] KNVD-IGT
Voorlichtingsdag Fotozetten,
Proc. RAI Congrescentrum, 19-3-1974.
Stencil.

- [17] Wong C.K. and Chandra A.K.,
Bounds for the String Editing Problem,
IBM Research Report, RC 4881, June 1974.
- [18] R. Morris and L.L. Cherry,
Computer Detection of Typographical Errors,
Bell Laboratories, CS Technical Report 18.
- [19] A.V. Hershey,
A Computer System for Scientific Typography,
Comp. Graphics and Image Processing (1972)1.
- [20] A.D. Hall,
SEdit; a Source Program Editor,
Bell Laboratories, CS Technical Report 116, May 1974.
- [21] B.R. Heap and M. Laws,
Construction of the Characters available in the NPL Graphical
Output System,
NPL Report CCU1, May 1968.
- [22] B.W. Kernighan and L.L. Cherry,
A System for Typesetting Mathematics,
Bell Laboratories, CS Technical Report 17.
- [23] J. W. Schoonaard,
User Oriented Text Editing:
Training Materials,
IBM Research Rep. RA 51, July 1973.
- [24] R. Reddy et al,
XCRIBL - a Hardcopy Scan Line Graphics System for Document
Generation,
Carnegie-Mellon Univ., Dept. of Comp. Sc. Report March,
October 1973.
- [25] P.J. Kalff Dr. (ed),
Tekstverwerking Ia, Systemen,
IGT, Amsterdam, 1973.
- [26] J. Kloppenburg and G. Veenker,
Der EDITOR, ein Programm zur Behandlung von Daten auf einem
Platten-Speicher,
Angewandte Informatik 4/1971.
- [27] R.A. Wisbey (ed),
The Computer in Literary and Linguistic Research,
Cambridge University Press, 1971.
- [28] H. Terai and K. Nakata,
Text Editing System Using On-line Real-time Character
Recognition.
Information Processing in Japan, 1974.
- [29] B.R. Schneider,
The Production of Machine-Readable Text:
Some of the Variables.
- [30] M.P. Barnett
SNAP - a Programming Language for Humanists,
The Computer and the Humanitas 4/1970.

- [31] P. Steuber,
A Demonstration of magazine page layout using a graphic display terminal,
In: Parslow R.D. and Green R.E. (eds)
Advanced Computer Graphics,
Plenum Press, London, 1972.
- [32] W.J. Hansen,
Graphic Editing of Structural Text,
In: Parslow R.D. and Green R.E. (eds)
Advanced Computer Graphics,
Plenum Press, London, 1971.
- [33] V.M. Briabrin,
On-line Text Editing using Visual Display,
In: Parslow R.D. and Green R.E. (eds)
Advanced Computer Graphics,
Plenum Press, London, 1971.
- [34] Z. Beer,
EDSYS.
Technical Report no 73-6,
Dept. of CS. Hebrew University, Jerusalem, may 1973.
- [35] Th. Pavlides,
Techniques for Optimal Compaction of Pictures and Maps,
Comp. Graphics and Image Processing, (1974)3.
- [36] A.J. Benjamin,
An Extensible Editor for a Small Machine with Disk Storage,
CACM, Vol 15, no 8 august 1972.
- [37] P. Hazel,
A General-purpose Text Editor for OS/360,
Software-practice and Experience, Vol 4, (1974).
- [38] A. Tepermann and J. Katzenelson,
A Format Editor,
Software - Practise and Experience, Vol 2, (1972).
- [39] R.A. Wagner,
Common Phrases and Minimum-space Text Storage,
CACM, Vol 16, no 3 march 1973.
- [40] J.P. McCarthy,
Automatic File Compression,
In: Guenther A. et al (eds)
International Computing Symposium 1973,
North Holland Publ. Co., 1974.
- [41] A. van Dam,
Hypertext,
In: Faiman and Nievergeld
Pertinent Concepts in Computer Graphics.
- [42] P.J.W. ten Hagen et al,
Design of an Interactive Graphics System,
MC Rapport IW 36/75.
- [43] W. Wakker,
Editors, een litteratuur onderzoek,
MC Memorandum, maart 1975.

5. Inhoud.

0. Inleiding.	1
0.1. Een Bedrijfssysteem voor Tekstverwerking	1
0.2. Een Systeem voor Ongeschoolde Gebruikers	2
0.3. Opslag van Teksten	3
0.4. Invoer en Uitvoer van Teksten.	4
1. Doel van het Tekstverwerkend Systeem.	5
1.1. Exploitatie en Onderzoek	5
1.2. Tekstverwerking voor Publikaties	5
1.2.1. Teksteditor speciaal geschikt voor het intypen van manuscripten	5
1.2.2. Proefdrukprogramma	6
1.2.3. Layout-programma's	6
1.2.4. Korrektieprogramma's	7
1.2.5. Speciale teksten	7
1.3. Tekstverwerking en Computer Graphics	7
1.4. Onderzoek naar Visuele Hulpmiddelen om Betekenis te Benadrukken	8
1.5. Tekstanalyse	9
2. Gebruik van de PDP8/I en PDP8/E Computers	10
2.1. Hardware	10
2.1.1. Thans aanwezige en binnenkort te verwachten apparatuur	10
2.1.2. Terminals	11
2.2. Programmatuur	12
2.2.1. Bedrijfssysteem	12
2.2.2. Filestysteem	15
2.2.3. Editor	16
2.3. Gebruikte Programmeertalen	16
3. Structuur	17
3.1. Het Onderste Nivo	18
3.2. Multi-User Editor en Filestysteem	19
3.3. Niet-Lineaire Teksten	21
3.4. Interface met het Grafische Systeem	21
3.5. Applicatieprogramma's	22
3.5.1. Layout-programma's	22
3.5.2. Afdruk- en Conversie-Programma's	23
3.5.3. Index- en Sorteert-Programma's	23
3.5.4. Tekstanalyse	24
3.5.5. Beheer en Ontwerp van Symboolverzamelingen	24
3.6. Fasering en Planning	24
4. Litteratuur	26
5. Inhoud	29

ONTYANGEN 2 8 SEP. 1978