
Nederlandstalige T_EX Gebruikersgroep

Aan : Leden Nederlandstalige T_EX Gebruikersgroep,
: Secretariaten T_EX Gebruikersgroepen,
: Europese T_EX coördinator.

Betreft : Verslag 6^e NTG bijeenkomst

Locatie : Digital Equipment bv, Utrecht

Datum : 20 november 1990

Behandeling : bij de volgende bijeenkomst (2 mei 1991 te Amsterdam)

Agenda:

1.	Opening	3
2.	Verslag bijeenkomst 11 mei 1990	3
3.	Ingekomen stukken en Mededelingen	4
4.	Stand van zaken Europese samenwerking/TUG international	5
5.	Verslag/discussie werkgroepen	6
6.	Verenigingszaken	9
7.	Rondvraag	9
8.	SGML (T _E X and ...)	9
9.	Sluiting	10

Bijlagen:

A	T _E X kalender	10
B	Werkgroepen NTG	11
C	Jaarverslag NTG 1990	13
D	Financiële verslag 1990	15
E	Concept begroting 1991	17
F	T _E X-NL subscription	18
G	NTG fileserver faciliteiten	21
H	Van de voorzitter: T _E X Nationaal en Internationaal	27
I	WG 1: Education	29
J	WG 4: Fonts	33
K	WG 6: Lijst en link met fotozetters	37
L	WG 7: PC-zaken; T _E X voor MS/PC-DOS PC's	38
M	Brief aan PC werkgroep: Enige suggesties voor WG-PC's	41
N	WG 8: NTG conferentie	43
O	WG 10: SGML-T _E X	44
P	Gebruik van T _E X binnen het EGD	47
Q	Gebruik van T _E X en L ^A T _E X op het CAWCS	48
R	Math into BLUes	57
S	Babel, a multilingual style -option system for use with L ^A T _E X's document styles	75
T	Frequently Asked Questions	85
U	Summary of Metafont Fonts Available	93
V	Program text generation with T _E X/L ^A T _E X	99
W	L ^A T _E X for enigneers and scientists (book review)	107
X	The structure of the T _E X processor	109
Y	Comments on the Future of T_EX and METAFONT	113
Z	NTG ledeninformatie	119

De NTG vereniging

Voorzitter:	C.G. van der Laan, Rijksuniversiteit Groningen. Internet: cgl@rug.nl
Secretaris:	G.J.H. van Nes, ECN, Service Unit Informatica, Petten. Internet: vannes@ecn.nl
Penningmeester:	J.L. Braams, PTT Dr Neher Laboratorium, Leidschendam. Internet: jl_braams@ptrnl.nl
Bestuursleden:	H.P.A. Mulders, KUB, Tilburg. T. de Klerk, DEC, Utrecht.
Postadres:	Nederlandstalige T _E X Gebruikersgroep, Postbus 394, 1740 AJ Schagen.
Postgiro:	1306238, t.n.v. Penningmeester NTG, Zoetermeer.
Internet:	ntg@hearn

De Nederlandstalige T_EX Gebruikersgroep (NTG) is een vereniging die tot doel heeft het bevorderen van de kennis en het gebruik van T_EX.

De NTG tracht dat te bereiken door het uitwisselen van informatie, het organiseren van congressen, symposia en tentoonstellingen m.b.t. T_EX en 'T_EX-produkten', en door het onderzoeken en vergelijken van T_EX met soortgelijke/aanverwante produkten, b.v. SGML.

De NTG biedt haar leden ondermeer het volgende:

- Tweemaal per jaar een NTG-bijeenkomst.
- Eenmaal per jaar open 'NTG-dagen', waar naast lezingen, ook cursussen (speciaal tarief voor leden) worden gegeven.
- De fileservers T_EX-NL waarop algemeen te gebruiken 'T_EX-produkten' staan. De meeste van deze T_EX-produkten zijn, tegen geringe vergoeding, ook op diskette verkrijgbaar. Daaronder valt ook een volledige MS-DOS versie van T_EX, L^AT_EX, en een previewer.
- De discussielijst T_EX-NL waarop vragen gesteld worden. Ook worden er via deze listserver ervaringen uitgewisseld.
- Activiteiten in werkgroepen.
- Eenmaal per jaar een ledenlijst met per lid informatie welke software en welke hardware, in relatie met T_EX, wordt gebruikt.

Lid worden kan door overmaking aan de penningmeester van het verschuldigde contributie bedrag. Daarnaast dient een informatieformulier te worden ingevuld, welke laatste via het secretariaat te verkrijgen is.

De contributie voor een persoonlijk lidmaatschap bedraagt *f* 75,-, de contributie voor een instituutslidmaatschap bedraagt *f* 200,-. Een instituutslidmaatschap geeft het recht om drie personen aan te wijzen die informatie welke aan de leden wordt verstuurd, ontvangen. Van die drie personen dient één persoon te worden aangewezen als rechtsgeldige vertegenwoordiger van het bedrijf/instituut, een ander als vervangend vertegenwoordiger.

Indien meer leden per bedrijf/instituut lid willen worden, geldt als additioneel tarief *f* 50,- per persoon. Voor studenten geldt eveneens een tarief van *f* 50,- (geen stemrecht). Voor afwijkende regelingen dient contact met het bestuur opgenomen te worden.

De statuten van de Nederlandstalige T_EX Gebruikersgroep zijn via het secretariaat te verkrijgen.

Nederlandstalige T_EX Gebruikersgroep

- Aanwezig** : E. Algera (EGD); A.W.W.M. Biegstraten (TUD); J. Braams (PTT Neher Laboratorium); F. D. Bijleveld (SWOV); N. Cox (KUN); M. Dings (AKZO Research); E.J. Evers (RUU); E. van Eynde (K.U. Leuven); C.M. Fortuin (Hogeschool Gelderland); Mevr. M. van Geest (CAWCS); J. van Gent (KUB); J.M. den Haan (TUD); J.B.W. Hoebeek (TUD); S. Hogeveen (Sterrenkundig Inst.); T.A. Jurriens (RUG); W.J. Karman (KUN); T. de Klerk (Digital); G.D.C. Kuiken (Kuiken); C.G. van der Laan (RUG); A. de Leeuw van Weenen (RUL); H. van der Meer (UvA); H.P.A. Mulders (KUB); G.J.H. van Nes (ECN); G. Oomen (Wolters Kluwer); P. van Oostrum (RUU); P. Vanoverbeke; N.A.F.M. Poppelier (Elsevier Science Publishers); C.H. de Ridder (Philips Components); P.B.M. Roes (Int. Statistical Inst.); K. Schilder (EGD); R. Smedinga (RUG); A. Soos (UT); A. Urvantsev (Rekencentrum Novosibirsk); E.J. Vens (RUG); J.H. Verniers (RAET); E. R. de Vreede (DCMR); J.J. Winnink; Mevr. D. van Wijnen (Wolters Kluwer).
- Notulist** : G.J.H. van Nes (ECN)

1 Opening

Voorzitter van der Laan opent om 10:10 uur de bijeenkomst en heet iedereen en in het bijzonder de Russische gast Anatoly Urvantsev, van harte welkom. Medegedeeld wordt dat gast op het eind van de bijeenkomst een korte schets zal geven van de lokale automatiseringswerkzaamheden.

De voorzitter benadrukt dat de NTG vergaderingen een bron zijn van onderlinge informele informatie uitwisseling hetgeen blijkt uit de geanimeerde discussies vooraf, gedurende de pauzes en gedurende de lunch. Ook de aanwezige uitgebreide leestafel blijkt duidelijk in de behoefte te voorzien.

Medegedeeld wordt dat de NTG vandaag geheel te gast is bij Digital Equipment B.V.. Het bestuur van de NTG wordt vervolgens voorgesteld.

2 Verslag bijeenkomst 11 mei 1990

Poppelier merkt op dat hij en enkele van zijn collega's het verslag en de convocatie graag eerder in zijn bezit willen hebben. Geantwoord wordt hierop dat de late datum van verschijnen mede het gevolg is van zowel het opnemen van de bijdragen van de afgelopen NTG conferentie als van de beschikbare tijd van de vrijwilligers die hieraan hebben meegewerkt. Een voordeel van het late verschijnen was ook het opnemen van zeer recent materiaal. Het uiteindelijke resultaat heeft positieve waardering van een aantal LUG's (Local User Groups) gekregen.

Besloten wordt om te trachten een conceptverslag bin-

nen een maand na de vergadering via electronic mail en/of via de post (voor degenen die niet via het netwerk zijn aangesloten) aan de NTG leden te sturen met daarbij de vermelding van de datum van de volgende vergadering. Het verslag kan vervolgens met correcties en/of aanvullingen opgenomen worden in het gedrukte document welke, samen met de uitnodiging, een maand voor de vergadering verstuurd zal gaan worden.

Betreffende het verslag van de 5e NTG bijeenkomst te Nijmegen, worden de volgende opmerkingen gemaakt (een aantal ervan zijn schriftelijk ontvangen):

- blz. 3: E.J. Vens (RUG) was op 11 mei niet aanwezig,
- blz. 3, 2e kolom: '...te gast bij Informatica en Computer & Communicatiezaken moet worden: '...te gast bij Wiskunde, Informatica, en Computer- & Communicatiezaken (C&CZ)',
- blz. 4, 2e kolom: 'De aanwezige worden gevraagd...' moet worden; 'De aanwezige leden worden gevraagd...'
- blz. 5, 1e kolom (WG1): 'Van Eijkhout zal ... boek verschijnen' moet worden: 'Eijkhout heeft contact met een uitgever over een door hem geschreven T_EX boek. Het zal niet voor 1991 verschijnen.'
- blz. 5, 1e kolom (WG3): 'Braams deelt mede dat bij zijn bedrijf iemand naar zo'n converter heeft gekeken' moet worden: 'Braams deelt mede dat bij zijn bedrijf iemand zo'n converter nodig had',
- blz. 5, 2e kolom (WG7): 'Bison (KUN) meldt dat...bezig' moet zijn: Bison (KUN) is met de em T_EX implementatie bezig,
- blz. 6, 1e kolom (WG8): '...onlangs een cadeau is gegeven' moet worden: '...onlangs een cadeau is ge-

This report has been generated by L^AT_EX software on SUN platform, using times.sty and multicol.sty (Frank Mittelbach), and a 300 dpi PostScript outputdevice. Editor of this report is G.J.H. van Nes.

Het verslag van de NTG bijeenkomst op 20 november 1990 is (in concept) eind januari 1991 via e-mail dan wel via de post reeds toegestuurd naar alle NTG leden.

stuurd',

- blz. 7, 1e kolom (WG13): Het artikel babel zal in een latere issue dan 11.3 van TUGboat verschijnen,
- blz. 8, 1e kolom (WG15): Vens (RUU) moet zijn Vens (RUG).

Het verslag wordt vervolgens goedgekeurd.

Betreffende de bijlagen wordt het volgende opgemerkt/medegedeeld:

- **Bijlage B:** De Amerikaanse TUG bijeenkomst is van 15–18 juli 1991 in Dedham (Massachusetts). Datum 7e NTG bijeenkomst is nog niet bekend.
- **Bijlage D:** Nr. 6: Auteur is Huub Mulders.
- **Bijlage E:** Listing van uucp en internet email adressen in hoofdletters verdient niet de voorkeur (zeker niet in een UNIX omgeving). Ook adressen die een machinenaam in zich hebben, worden afgeraden.
- **Bijlage G:** Het e-mail adres van Piet van Oostrum moet zijn: piet@cs.ruu.nl
- **Bijlage AA:** De DECUS tape is nog niet ontvangen. Na ontvangst zullen de tijdens de vergadering ingeleverde tapes beschreven en vervolgens geretourneerd worden.
- **Bijlage CC:** Verwijzingen in de literatuurlijst zijn niet in orde voor die welke ook in de daarop volgende bijlage voorkomen. Ze komen dan overeen met de nummering daarin. Of de oorzaak bij \LaTeX ligt is (nog) onbekend.
- **Bijlage EE:** De ovalen in figuur 1 en 2 blijken een fout te hebben (vershoven horizontale lijn) welke echter afhankelijk is van de gebruikte versie van \LaTeX / \TeX . Sommige implementaties (o.a. DEC) blijken daarentegen een vershoven verticale lijn te bevatten. Oude \LaTeX versies hebben het probleem niet evenals de \LaTeX versie van januari 1990. Braams denkt de oorzaak te kennen. Poppelier zal een reparatiepoging ondernemen. Genoemd wordt dat bij gebruik van \TeX CAD dezelfde fout werd geconstateerd.
- **Bijlage HH:** Een aantal correcties welke van de NTG leden zijn ontvangen, zullen in de volgende lijst opgenomen worden. De voorzitter stelt de vraag of er bezwaar is om de NTG adressen aan derden door te geven voor niet commerciële doelen. Een eerste verzoek voor het bestand is reeds door de Franse usergroep (GUTenberg) ontvangen (dit voor verzending van Euro \TeX 91 annonces etc.). Genoemd wordt de wet persoonsregistratie. Een mogelijke optie is om de vraag op het aanmeldingsformulier op te nemen. Besloten wordt om het probleem aan de juridische expert van de SGML Users Group (Koen Mulder) voor te leggen. Vervolgens zal het bestuur een voorstel doen.

Gevraagd wordt tenslotte om nieuwe bijdragen voor bij het volgende verslag. Gedacht wordt aan ondermeer algemene stukken m.b.t. de acceptatie van \TeX bij bedrijven.

Poppelier deelt mede een \LaTeX boekbespreking te verzorgen.

3 Ingekomen stukken en Mededelingen

De volgende stukken zijn ingekomen:

- Afschrift van de brief van V. Eijkhout, ontvangen via de SGML- \TeX penningmeester, betreffende de logistiek tijdens zijn cursus. De brief is door het NTG bestuur als kennisgeving aangenomen. Een antwoord op de brief is verstuurd door het SGML bestuur. Op dringend verzoek van Poppelier wordt op de inhoud ingegaan bij agendapunt 5 (NTG werkgroep 8).

- Van Jos Winnink werd de volgende email ontvangen waarin de distributie van \TeX software ter sprake komt:

"Ik vraag mij af of het in internationaal verband mogelijk en zinvol is om te komen tot een constructie met betrekking tot de distributie van \TeX programmatuur zoals die geldt voor de SIMTEL20 archieven. De daar gebruikte constructie is als volgt:

De SIMTEL20 archieven bevinden zich in een computer in de USA. In Europa bestaat een systeem van TRICKLE servers, die files cachen. De complete directory is altijd bereikbaar via de TRICKLE servers. Bij aanvraag van een file kijken de TRICKLE servers eerst in hun eigen bestand, dan wordt gekeken of de file in EUROPA bij één van de TRICKLE servers is en mocht dit niet zo zijn dan wordt de file aangevraagd in de V.S.

Het voordeel van dit systeem is dat vanuit de gebruiker gezien er slechts één centrale plaats is waar hij/zij software aanvraagt en dat de gebruiker zich niet hoeft te bekommeren op welke manier of via welk netwerk het bestand uiteindelijk via EARN tot hem/haar komt."

Op deze zogenaamde 'intelligente (virtuele) servers' benadering wordt gereageerd door de voorzitter. De aanpak zoals door Winnink wordt genoemd is binnen het bestuur reeds eerder besproken. Het zou een uitstekende manier zijn van optimale 'fileservers architectuur'. Het TRICKLE systeem schijnt echter nog niet volledig goed te werken. Een opdracht voor verder onderzoek in deze richting wordt neergelegd bij de werkgroep 'communicatie'. Het zal ook een van de eerste taken kunnen zijn bij een internationale samenwerking van de Europese \TeX gebruikersgroepen.

- Magazines van zusterverenigingen:
 - SGML-Bulletin Holland
 - Cahiers Gutenberg
 - DANTE \TeX nische Komödie
 - \TeX line (Malcolm Clark)

Deze documenten bevinden zich op de leestafel. Informatie erover is op te vragen via het NTG secretariaat.

- AMS guidelines.
- Aankondigingen \TeX Usermeetings (zie bijlage B van het vorige verslag).

- Hope Hamilton's NCAR sjablonen/templates van \TeX codes. Melding hiervan vond reeds plaats op de vorige bijeenkomst. Het materiaal is nu beschikbaar als listing en als MS-DOS diskette. Jurriens (RUG) deelt mede dat hij de code zal onderzoeken op nuttigheid. Relevante zaken kunnen dan later beschikbaar worden gesteld.
- Anita Hoover WordPerfect-5 interface macro's.

De volgende mededelingen worden gedaan:

- **Bestuursvergaderingen.**
Er hebben 2 telefonische bestuursvergaderingen plaatsgevonden sinds de 11 mei NTG bijeenkomst.
- **NTG ledenaantal.**
Sinds de vorige vergadering is het NTG ledenaantal weer toegenomen en wel met 25 nieuwe leden, inclusief 5 extra instituutleden. De NTG telt nu in totaal 117 leden (waaronder 20 instituutleden).
- **TeX-NL listserver.**
De TeX-NL listserver groeide van 100 deelnemers in mei tot 117 in november.
- **Contributiebetalingen.**
De penningmeester deelt mede dat bij betaling van contributie veelal de naam van de persoon en/of bedrijf ontbreekt hetgeen tot moeilijkheden leidt bij de verwerking. Bij de contributieverwerking voor 1991 zal er met referentienummers gewerkt worden welke ook bij de betaling vermeld moeten worden.
- **TeX 3.0 distributie.**
Genoemd wordt de \TeX 3.0 software distributie: de DECUS tape die tijdens de vergadering ingeleverd kan worden, de nieuwe UNIX software en de MS-DOS en \TeX implementatie welke later op de vergadering ter sprake zal komen.
- **De leestafel.**
Gewezen wordt op de aanwezige uitgebreide leestafel. Hiervoor blijkt ook een duidelijke belangstelling te bestaan.
- **L^ATeX cursusmateriaal.**
Van Piet Tutelaers (TUE) is een email ontvangen waarin hij het volgende meedeelt:
"De eerste versie 'Sheets van de L^ATeX cursus' hebben we omgedoopt tot 'Cursus L^ATeX'. Een aantal sheets die ik met knippen en plakken en met de hand had gemaakt zijn nu ook in L^ATeX of in een andere elektronische vorm aanwezig. Ron Helwig (een collega) en ik willen deze nieuwe cursus beschikbaar stellen. Alleen weten we nog niet precies hoe. Het leukste zou zijn in elektronische vorm. Wij delen het uit op de cursus als een losbladige A4 map."
- **Ledenwervingsactiviteiten.**
Genoemd wordt het feit dat er slechts drie leden/instituten uit de wiskunde hoek lid zijn van de NTG. Het beleid van het NTG bestuur zal ondermeer zijn om deze groep actief te benaderen. Daarom zal contact worden opgenomen met zowel het Wiskundig Genootschap als het Centrum voor Wiskunde en Informatica (CWI). Nagegaan zal worden of beide partijen wederzijds iets voor elkaar kunnen beteke-

nen analoog aan de hechte band in Amerika tussen de American Mathematical Society (AMS) en TUG. Gevraagd wordt om suggesties bij de NTG leden.

Het formeren van een werkgroep in deze zou mogelijk niet efficiënt zijn. Daarentegen zouden wel enkele 'PR-leden' bepaalde activiteiten kunnen ontplooiën. Eerste prioriteit van de NTG ligt in Nederland bij de wetenschappelijke markt en in het bijzonder bij de wiskundigen. \TeX kent binnen deze groep een zeer belangrijke plaats (typografie) die bijvoorbeeld door WordPerfect niet ingevuld kan worden.

De Leeuw van Weenen benadrukt dat NTG leden en \TeX gebruikers zeker niet dezelfde groep zijn. Binnen de mathematische afdeling van de Universiteit van Leiden is slechts een enkel persoon NTG lid doch \TeX wordt daarentegen zeer intensief gebruikt door zowel staf als afstudeerders. Opgemerkt wordt ook dat veel \TeX gebruikers die het systeem goed kennen (en/of gebruiken) en daarbij geen behoefte hebben aan aanvullende informatie, het niet direct noodzakelijk vinden om zelf ook lid te worden van een \TeX gebruikersgroep.

Een discussie vindt vervolgens plaats of er een algemeen, dan wel een op maat gesneden verhaal beschikbaar moet komen waarin de plaats van \TeX binnen de documentpreparatie systemen duidelijk naar voren komt.

Tenslotte wordt opgemerkt dat bij congressen (zoals de NTG dagen) wel de positie en de functie van \TeX duidelijk naar voren moet komen. In relatie daarmee kunnen ook artikelen geschreven worden bijvoorbeeld t.b.v. het Wiskundig Genootschap.

Winnink stelt voor om de basis- en middelbare scholen kennis te laten maken met \TeX , daar bij deze groepen steeds meer de PC zijn intrede aan het doen is. De scholen zouden een goede basis voor het toekomstig gebruik van \TeX kunnen zijn.

De voorzitter antwoordt hierop dat bij de afgelopen Cork90 presentatie op deze gestelde vraag een nee werd geantwoord. Daarentegen is men in Duitsland (DANTE) wel binnen het middelbaar onderwijs bezig. Het is echter beter om eerst hun resultaat af te wachten om daarbij van eventueel gemaakte fouten te leren.

Winnink wijst op het Comenius project. Hij zal kijken of het mogelijk en zinvol is om op individuele basis enige activiteiten te ondernemen.

4 Stand van zaken Europese samenwerking/TUG internationaal

De voorzitter geeft een uiteenzetting over de huidige situatie bij TUG (de Amerikaanse \TeX Gebruikersgroep). Een eerste verslag is reeds te vinden als bijlage X van het verslag van de vorige 11 mei bijeenkomst.

Binnen de TUG organisatie zijn enige problemen ontstaan zowel op organisatorisch, persoonlijk als op finan-

cieel gebied. De uitvoerende (bezoldigde) directeur Ray Goucher is weg uit het TUG kantoor vooral vanwege onenigheid met de TUG Board of Directors (BOD). Een begrotingstekort is aanwezig, ons inziens vanwege de hoge personele lasten (directeurschap).

Genoemd wordt ook dat de kracht van de nationale werkgroepen debet aan de moeilijkheden bij TUG zijn. LUG's (Local User Groups) als die in Europa en in de toekomst zeker ook die in Japan en de Oost Europese landen, noodzaken tot een volledig hernieuwde bestuurlijke vorm van de Amerikaanse T_EX Gebruikersgroep. Activiteiten die ontwikkeld worden t.a.v. T_EX zaken zijn op dit moment, i.t.t. enige jaren geleden, zelfs groter in Europa dan in Amerika.

Een machtsstrijd is niet alleen in Amerika gaande. De invloed die een grote (en nog steeds groeiende) LUG als DANTE wil hebben op TUG is zeker niet te bagateliseren.

De mening van het volledige NTG bestuur, welke tevens door de aanwezigen wordt gedeeld, is dat TUG bij een democratische besluitvorming niet van de grootte van LUGs mag uitgaan. TUG overleg zou alleen moeten plaatsvinden met een enkel vertegenwoordiger per land, waardoor het gewicht van elke LUG gelijk is. Genoemd wordt het verschil in Amerikaans en Europees verenigingsrecht.

Er zijn onlangs enkele commissies voorgesteld om de TUG zaken in de negentiger jaren in betere banen te leiden:

- 'Transition Management Committee'; om het kantoor weer op poten te zetten,
- 'Long Range Planning Committee'; had allang ingesteld moeten worden,
- 'Publication Planning Committee',
- 'Character-Encoding Committee'.

Op publicatie gebied wordt gedacht aan twee soorten media: TUGboat met algemene TUG onderwerpen (kwaliteit), en een internationale newsletter met locale edities (taal aanpassingen c.q. uitbreidingen.) De laatste zou vooral portable, flexibel en snel moeten verschijnen.

5 Verslag/discussie werkgroepen

Verslag wordt gedaan van de activiteiten binnen een aantal NTG werkgroepen. De volgorde van de werkgroepen binnen dit verslag is niet overeenkomend met de volgorde van behandeling gedurende de vergadering welke laatste gebaseerd was op belangrijkheid en de beschikbare informatie van het onderwerp.

Werkgroep 1: Cursusmateriaal

Van der Laan deelt mede dat hij met een artikel over cursussen bezig is. Daar het geschrift nog niet af is, wil hij dit onderwerp verschuiven naar de volgende vergadering.

Werkgroep 2: Richtlijnen voor publicaties/conferentie proceedings

Jurriens deelt mede dat hij vooral bezig is zijn eigen mensen te instrueren bij het leveren van bijdragen voor artikelen en boeken.

Poppelier vindt deze werkgroep niet nodig daar het maken van richtlijnen meer het werk is voor uitgeverijen; zij horen met initiatieven en richtlijnen te komen. Hij stelt daarom ook voor om de werkgroep op te heffen. Na een mondelinge stemming (2 tegen en 4 onthoudingen) wordt aan het voorstel voldaan.

Werkgroep 4: Fonts

Vens neemt zitting in de werkgroep. Hij deelt mede dat de lijst van deelnemers van de werkgroep nodig opgeschoond moet worden. Doelstelling van de werkgroep moet vervolgens geformuleerd worden. Onderwerpen zijn ondermeer de naamgeving van fonts, inventarisatie van wat er wel/niet is, regelen van de centrale/decentrale beschikbaarstelling van fontbestanden, contacten onderhouden met fonts expert in het buitenland. Ook moet de werkgroep het Nederlands(talig)e contactpunt zijn voor Amerika. Winnink meldt zich tevens aan voor de werkgroep.

Werkgroep 5:

Drivers, previewers, printers, postscript

De werkgroep had niets te melden. Winnink deelt mede dat het beschikbaar stellen van een lijst van beschikbare output drivers belangrijk is. Daarbij zou ook de systemen vermeld moeten worden waarop de drivers operationeel zijn. (Zie echter ook TUGboat, met Don Hoseks overzichten.)

Werkgroep 6: Lijst en link met fotozetters

Jurriens verzoekt degenen die ervaring hebben met (fotozetter) bedrijven, contact met hem op te nemen. Zijn ervaring is dat indien men een eigen fotozetter aan T_EX heeft kunnen koppelen het uiteindelijk aan tijd/interesse ontbreekt om dat ook aan anderen kenbaar te maken. Jurriens zal proberen om de volgende vergadering met enig resultaat te komen.

Hij deelt tevens mede dat de Universiteitsdrukkerij van de RUG, uitgaande van .dvi files, output in 600 dpi kan aanleveren voor *f.* 1.50 per bladzijde.

Werkgroep 7: PC-zaken

Winnink geeft een toelichting m.b.t. bijlage J van het vorige verslag. Hij noemt het voordeel van de TE editor op de Gutenberg diskettes vooral voor het herstellen van fouten in programma's (twee window systeem). Missende in het onderzoek van Winnink is de PC-T_EX

implementatie. Bij gebruik van PC- \TeX is het niet mogelijk om \LaTeX te gebruiken als PCSA geladen is. Wanneer em \TeX onder dezelfde omstandigheden wordt gebruikt kan \LaTeX wel worden gebruikt. Mulder deelt mede dat de performance tussen PC- \TeX en em \TeX ruwweg gelijk is.

N.a.v. een vraag over de beschikbaarheid van de MS-DOS em \TeX versie antwoordt Winnink dat hij eind november em \TeX in de vorm van diskettes zal rondsturen naar degenen die in deze implementatie geïnteresseerd zijn. De software bestaat uit 15 diskettes van 1.44 Mbytes. Daar belangstellenden zelf een kopie van de moederversie moeten maken is het systeem redelijk goed tegen virussen beschermt.

Naar voren gebrachte alternatieven als het extern kopiëren via het TUG kopieersysteem dan wel via een commerciële instellingen bieden weinig voordelen. Ook De Ridder (Philips) en Vens (RUG) zijn genegen voor kopieën te zorgen (Bison/KUN heeft zich al eerder voor het uitvoeren van deze taak beschikbaar gesteld).

Van der Meer informeert naar een gelijksoortig onderzoek m.b.t. de Atari Macintosh implementatie. Geantwoord wordt dat de Atari systemen al eerder (doch in veel mindere mate) bekeken zijn.

Algera meldt dat hij binnenkort een Public Domain versie voor Amiga systemen uit Duitsland hoopt te ontvangen.

Poppelier deelt mede dat de doos met Atari diskettes, eerder afkomstig van Matthius Moritz (zie vorige verslagen), nu bij hem in bezit zijn. Het bevat echter veel doublures en oude versies van WEB en METAFONT. Door Sonnemans (Atari-ST auteur) is wel eerder belangstelling getoond voor de verzorging van de gehele Atari implementatie, doch tot heden is niets meer van hem vernomen.

Op de vraag van de voorzitter of de aanwezigen interesse hebben in een verder onderzoek van Atari PD software wordt negatief geantwoord.

Winnink deelt mede dat indien voor alle systeemplatforms goede Public Domain implementaties beschikbaar zijn, de werkgroep pas een slapend leven kan beginnen. Opgemerkt wordt het belang van dergelijke onderzoeken t.b.v. publicatie in TUGboat. Winnink zal hiervoor actie ondernemen.

Genoemd wordt ook de mogelijkheid om PD versies op te nemen in het nog samen te stellen NTG informatie pakket. Het bestuur denkt aan het opnemen van een MS-DOS demo versie.

Werkgroep 8: Nederlandstalige \TeX gebruikersdag

Verwezen wordt naar de bijlagen M t/m T behorende bij het verslag van de vorige NTG bijeenkomst. De organisatie wordt bedankt voor de geslaagde dagen.

Op aandringen van Poppelier komt de aan de SGML-

\TeX organisatie gestuurde brief van één van de docenten ter sprake. De docent Eijkhout vermeldt daarin een drietal zaken die volgens hem niet juist waren geregeld. Na een uitgebreide uiteenzetting door de voorzitter over de juiste toedracht, besluit één van de cursisten, Cox, de discussie met de opmerking dat hij van enig ongemak niets had gemerkt en voor hem de cursus zeer goed was verlopen.

Daar de penningmeester nog geen financiële afrekening heeft ontvangen, kunnen de huidige leden van de werkgroep nog niet gedechargeerd worden. Dit punt wordt dan ook naar de volgende vergadering verschoven.

Bij het organiseren van de NTG dagen in 1991 wordt gedacht aan de KUB, gastheer Joop van Gent en Huub Mulders.

Van Gent staat duidelijk niet te popelen om medewerking te verlenen aan de organisatie van de nieuwe NTG dagen, doch hij ziet wel in dat het doorgaan van het gebeuren afhankelijk is van een aantal vrijwilligers. Hij licht toe dat één van voorwaarden om een conferentie te organiseren is, dat niet alleen gesproken zal gaan worden over \TeX doch meer over documentsystemen in het algemeen, t.w. document-productie en document-retrieval. Of een conferentie over dit algemene onderwerp doorgaat is afhankelijk van het al dan niet goedkeuren van een onderzoeksvoorstel bij zijn werkgever. Een beslissing hierover is binnen enkele weken te verwachten. Alleen bij een positieve beslissing ziet hij duidelijk kans om ook tijd vrij te maken voor het organiseren van de NTG dagen in 1991.

De voorzitter benadrukt dat het meeste werk bij het organiseren van de NTG dagen, het vinden van de juiste sprekers is. Daarnaast dienen in een zeer vroeg stadium zaalreserveringen plaats te vinden. Dit laatste zal volgens van Gent geen problemen opleveren.

Genoemd wordt dat een belangrijke belemmerende factor bij het organiseren van conferenties bij een vereniging als de NTG, de bereidwilligheid is van de vrijwilligers, vooral van degene die de voortrekkersrol (verantwoordelijkheid) toebedeeld krijgt.

Van Gent acht de kans klein dat het projectvoorstel, en daarmee gekoppeld de organisatie van deze interessante algemene conferentie over documentsystemen, geen doorgang zal vinden. Hij wil echter geen beloftes op dit moment doen. Discussie vindt vervolgens plaats over het al dan niet tijdens de vergadering te benoemen van een voorbereidingscommissie (= nieuwe leden werkgroep 8), het al dan niet laten doorgaan van de NTG dagen in 1991, en het verplaatsen van de conferentiedagen naar 1992.

Besloten wordt om als nieuwe leden van werkgroep 8 te benoemen van Gent (KUB), Mulders (KUB) en Jurriens (RUG). Jurriens nam ook deel aan de organisatie van de NTG dagen in 1990 en zorgt dus voor de continuïteit. Genoemde personen zullen proberen de komende conferentie voor te bereiden. Indien dit niet mocht lukken zal

de organisatie van de NTG dagen op de komende (mei) vergadering opnieuw ter sprake worden gebracht.

Werkgroep 9: Integratie beelden en \TeX

Jurriens deelt mede dat pogingen zijn gedaan om bijeen te komen doch dat hiervan nog geen realisatie heeft plaatsgevonden. Hij heeft zelf ervaring met PiCTeX . In relatie met emTeX heeft het bigTeX nodig. Er is van de werkgroep niet direct rapportage te verwachten. Aan de werkgroep wordt dringend gevraagd om een inventarisatie te maken van de aanwezige mogelijkheden. Daarnaast zal getracht worden een taakomschrijving te maken.

Werkgroep 10: SGML- \TeX relatie

Van der Laan treedt als coördinator van de werkgroep af. Hij deelt mede dat de werkgroep reeds veel gedaan en gepubliceerd heeft. Bijlage EE van het vorige verslag toont enig resultaat; daarnaast zijn een aantal rapporten verschenen en is er een conferentie over SGML- \TeX georganiseerd. Een rapport over tabellen wordt nog aangehouden daar het nog niet volledig is en de interesse over het onderwerp verslapt.

Werkgroep 13: Nederlandstalige \TeX

Gemeld wordt dat Eijkhout (Univ. Illinois) ondanks zijn verblijf in de USA, voortrekker zal blijven van deze werkgroep.

Het verslag van de werkgroep is als bijlage L opgenomen bij de notulen van de vorige vergadering. Belangrijkste activiteiten op dit moment zijn het onderhoud van de gemaakte stijlen en het beschikbaarstellen van de `.doc` stijlopties.

Winnink heeft problemen met de `dutch.sty` en `brief.sty` samen met \TeX 3.0 en language keuze `dutch=0`. Volgens hem dient de Nederlandse `.sty` files aangepast te worden i.v.m. de taal uitbreidingen in de nieuwe versie van \TeX . Hij zal informatie over eigen ervaringen naar Braams sturen waarna eventueel de nodige aanpassingen worden aangebracht.

De voorzitter deelt mede dat hij onlangs via een email van Veldhuyzen van Zanten (SARA) de vraag ontvangen heeft of vanaf \TeX versie 3.0 de namen 'ntex' en 'nlatex' (Nederlandstalige \TeX en \LaTeX versie) gehandhaafd blijven of dat men in plaats hiervan van `\language` gebruik moet gaan maken. Voordat hij de software bij de SARA wilt installeren, zou hij graag eerst de mening over dit punt van de NTG willen vernemen. Veel onduidelijkheid is er m.b.t. dit punt. Grootte van de hyphenation files zal een belangrijke factor worden. Daarnaast is flexibiliteit een belangrijk item. De vraag wordt doorspeeld naar de werkgroep.

Van der Meer stelt de werkgroep voor om diverse afbreekfiles samen te stellen waarbij grootte het belangrijkste is. Ter sprake komt de Nederlandstalige hyphenation

files t.w. die van Peter Vanroose (klein) en de CELEX file (groot). De Peter Vanroose file zou zeer beperkt zijn (verschillende meningen komen naar voren: van 'zeer goed te gebruiken' tot 'van geen waarde'). De CELEX file is aldus Kuiken echter beduidend te verkleinen door de 'vijfjes' te verwijderen. Winnink merkt op dat ook al worden op basis van een hyphenation file alle woorden *syntactisch* correct afgebroken, het dan nog nodig is om na te gaan of de afbrekingen gewenst zijn. Kuikens stelt voor om een tweede 'CELEX' te maken en deze ter beschikking te stellen.

De voorzitter concludeert dat het zinvol zou zijn om een en ander op schrift te stellen en dit als bijlage bij het volgende verslag op te nemen. Kuiken en Winnink zullen hiervoor zorgdragen. Opgeroepen wordt om ideeën aan hen door te geven.

Er zou op de NTG server zowel een goede minimale als een maximale file geplaatst moeten worden.

Werkgroep 14: Communicatie

Braams deelt mede dat via de werkgroep diverse files op de server zijn geplaatst.

Van Winnink heeft het bestuur een email ontvangen betreffende een algemene opgezette en uniforme wijze van het ontvangen van \TeX bestanden zoals deze ook bij TRICKLE en SIMPTTEL20 geldt (zie ook agendapunt 'Ingekomen stukken'). Kort wordt nog op dit onderwerp van uniforme benadering van een dichtbijzijnde server voor het verkrijgen van \TeX file bestanden, ingegaan. Braams vermeldt dat eerst alle servers op hetzelfde niveau gebracht moeten worden. Genoemd wordt daarbij ook enkele mogelijkheden van de TUGLIB server. Oostrum zal het onderwerp ter sprake brengen in de \TeX archive-mailing listserver teneinde de discussie daar (wederom) op gang te brengen. De NTG werkgroep 14 wordt ook gevraagd om de nodige acties te ondernemen.

Werkgroep 15: \TeX 3.0

Van Oostrum (RUU) meldt dat de huidige versie van \TeX en METAFONT respectievelijk 3.1 en 2.7 is (zie ook bijlage GG van het vorige verslag). Niet alle operating systemen hebben op dit moment dezelfde versie beschikbaar. Een overzicht wordt vervolgens gegeven van de huidige stand van zaken:

- **UNIX systemen:**

\TeX 3.1 en METAFONT 2.7 is via de Universiteit van Washington te bestellen, inclusief Web2c 5.8a. Van Oostrum heeft deze software op de RUU server geplaatst. De software is ook via Van Campen (KUN) te verkrijgen.

- **VMS systemen:**

De 1990 DECUS \TeX tape bevat \TeX 3.0 en METAFONT versie 2.0. De CHANGE files van 3.1 en 2.7 zijn via ftp te verkrijgen. Braams zal proberen deze files op de in Nederland te distribueren tape te plaatsen.

- **DOS implementaties:**

\TeX 3.0 (em \TeX en Sb \TeX) en METAFONT 2.0 (em \TeX).

- **Atari systemen:**

Men is elders bezig met de PD implementatie van \TeX 3.0. De versie loopt echter niet binnen 1 Mbyte. Het heeft een fontbibliotheek als em \TeX , doch er is geen compatibiliteit. Helaas is in deze geen standaard. Hogeveen meldt dat binnen 4 Mbyte een versie operationeel is. Hij probeert na zijn promotieonderzoek een versie voor een 1 Mbyte systeem te maken. Poppelier noemt een commerciële versie die begin 1991 verscheept zal worden en wel binnen 1 Mbyte werkt. Deze versie kent virtuele fonts.

- **Macintosh:**

De Oz \TeX versie 1.3 is nagenoeg op niveau \TeX 3.0. De versie is te verkrijgen bij van der Meer (UvA).

Poppelier benadrukt de belangrijkheid van het bekijken van de nieuwe features in \TeX 3.0. Daar in deze versie een mechanisme aanwezig is om verschillende talen binnen een enkel document te gebruiken, is standarisatie op 'language-gebied' hard nodig (zie ook het bespreken bij werkgroep 13). Daarbij moet ondermeer gedacht worden aan uniformiteit van de taal-commando's, (`\dutch?`, `\english?`), de functionaliteit van deze commando's (meer dan alleen `\frenchspacing`) en het al dan niet gebruik van taal-nummering.

Een voorstel wordt gedaan om één en ander in de `hyphen.tex` file zelf vast te leggen. Genoemd wordt het 'babel-project' (werkgroep 13; zie ook bijlage R van het vorige verslag). Gedacht wordt om zelf het voortouw te nemen bij het oplossen van dit multilanguage probleem en vervolgens de resultaten bij TUG te deponeren. Mogelijk kan dan daarbij snel tot acceptatie gekomen worden. (Opmerking toegevoegd: Michael Fergusson is de multilanguage coördinator, het is dus een goede zaak dat betrokkenen met hem erover van gedachten wisselen.) Zo gauw em \TeX bij Braams is geïnstalleerd, zal hij verdere acties ondernemen. Vervolgens wordt in het kort wat verteld over het babel-project. De `babel.sty` code zou op de NTG server geplaatst kunnen worden om zodoende wat meer bekendheid eraan te geven.

Voorgesteld wordt om in de komende mei vergadering, dieper op de genoemde onderwerpen in te gaan.

Winnink noemt enige problemen met de `brief.sty` in samengaan met de `dutch.sty` optie.

Mulders brengt de taalafhankelijkheid van METAFONT ter sprake. De nieuwe versie schijnt ingewikkelde ligaturen aan te kunnen.

6 Verenigingszaken

De voorzitter zet uiteen dat in de eerste helft van 1991 bestuursverkiezingen zullen worden gehouden. Het huidige bestuur dateert uit juni 1988, waarbij (om praktische

redenen) zij niet 'democratisch' gekozen zijn. Afgesproken is reeds dat jaarlijks twee bestuursleden zullen aftreden dan wel zich herkiesbaar zullen stellen.

Voor de verkiezingen in 1991 zullen aftreden de voorzitter in functie, van der Laan, die zich echter wel herkiesbaar stelt. Aftredend en niet herkiesbaar (vanwege stopzetting directe \TeX activiteiten bij werkgever; \TeX is wel verwerkt in Digitals produkt: VAX-document) is Theo de Klerk.

Het bestuur zal ruim voor de volgende vergadering met een voorstel van twee kandidaten komen. Daarnaast kunnen ook de huidige NTG leden nieuwe kandidaten voordragen (overeenstemmend met de statuten). Van kandidaats bestuursleden wordt wel een curriculum vitae gevraagd om zodoende de leden over de nieuwe kandidaten bij de verkiezingen te informeren. Alle NTG leden zullen nog via een separaat schrijven worden geïnformeerd.

Op de agenda van de volgende NTG vergadering zal ook de financiële zaken behandeld worden (naast vanzelfsprekend de noodzakelijke bijlage in het komende verslag).

7 Rondvraag

Hiervan wordt geen gebruik gemaakt.

8 SGML (\TeX and . . .)

Van der Laan (RUG) geeft een presentatie over SGML en \TeX . De tekst ervan is grotendeels opgenomen als bijlage EE van het vorige verslag.

De spreker deelt mede dat het presentatie materiaal (overhead sheets) beschikbaar is voor de NTG leden. Het materiaal is aanwezig in twee vormen: een verhaal voor managers en een technisch verhaal.

Op verzoek van één van de NTG leden geeft de aanwezige gast Anatoly Urvantsev, hoofd rekencentrum Novosibirsk (USSR), een kort overzicht van zijn automatiseringswerksituatie.

Op de volgende vergadering zal Eric Algera (Energiebedrijf voor Groningen en Drente) praten over hun lokale situatie m.b.t. het beschikbaarstellen en het gebruik van \LaTeX/\TeX .

9 Sluiting

De datum voor de volgende vergadering is:

donderdag 2 mei 1991

bij Elsevier Science Publishers te Amsterdam; gastheer Nico Poppelier.

Digital Equipment B.V. wordt bedankt voor de geboden gastvrijheid en diensten. De aanwezigen worden bedankt voor hun bijdragen in de discussie.

De vergadering wordt om 16:30 uur gesloten waarna een informele bijeenkomst plaatsvindt onder het genot van een door Digital Equipment aangeboden drankje.

Getekend:
Voorzitter:

Secretaris:

BIJLAGE A

T_EX kalender 1991

20–22	feb	Deutsche T _E X (10 ^e)	Wenen, Oostenrijk
2	mei	NTG (7 ^e)	Elsevier, Amsterdam
28–30	mei	GUTenberg'91	Parijs, Frankrijk
15–18	jul	TUG	Dedham, Massachusetts, USA
23-26	sep	EuroT _E X	Parijs, Frankrijk
	nov	NTG (8 ^e)	?

BIJLAGE B

Werkgroepen Nederlandstalige T_EX Gebruikersgroep

1. Educatie

A.W.W.M. Biegstraaten (TUD)

G. Haayer (Styx Publications)

C.G. van der Laan

(coördinator)

J.R. Luyten (RUG)

P. Tutelaers (TUE)

2. (werkgroep is vervallen)

3. Evaluatie produkten (Ned. L^AT_EX incl sty. files en afbreekregels; andere macrocollecties AMST_EX; converters K-talk; T_EX naar ASCII; index programmatuur; dBase-T_EX koppeling; adreslabels; verkrijgbaarheid etcetc.)

J.L. Braams (PTT Research Neher Lab)

(coördinator)

M.A.J.H. Broeren (Océ Nederland B.V.)

G. Haayer (Styx Publications)

J.R. Luyten (RUG)

H.P.A. Mulders (KUB)

4. Fonts (gebruik van Metafont)

H. Brouwer (EGD)

G. Haayer (Styx Publications)

(coördinator)

A.J. de Meyer (RUU; Wiskunde)

P. Tutelaers (TUE)

F.J. Velthuis (RUG; Rekencentrum)

J.C. de Moor (Theol. Univ.)

E.J. Vens (RUG)

J.J. Winnink

5. Drivers, previewers, printers, postscript

J.L. Braams (PTT Research Neher Lab)

H. Brouwer (EGD)

P. Tutelaers (TUE)

6. Lijst en link met fotozetters

G. Haayer (Styx Publications)

M. Helmig (Philips)

T.A. Jurriens (RUG; Sterrenkunde)

F.J. Velthuis (RUG; Rekencentrum)

(coördinator)

7. PC-perikelen; campuslicentie etc.

E. Algera (EGD; Amiga)

G.J. Braas (EGD; Archimedes)

H. Brouwer (EGD)

J.R. Luyten (RUG; DOS)

P. Tutelaers (TUE)

E.J. Vens (RUG; DOS)

J.J. Winnink (-; DOS)

E.B.J. van der Zalm (RUU; Atari)

R. Veldhuyzen van Zanten (SARA; McIntosh)

8. **Nederlandse T_EX gebruikersdag (1992)**
 J. van Gent (KUB)
 A. van Horck (KUB)
 H.P.A. Mulders (KUB)
 H. Paijmans (KUB)
 T.A. Jurriens (RUG)
9. **Integratie beelden en T_EX**
 H. Brouwer (EGD)
 G.D. Draaijer (MARIN)
 M. Helmig (Philips)
T.A. Jurriens (RUG; Sterrenkunde) (coördinator)
10. **SGML-T_EX relatie**
 A.W.W.M. Biegstraaten (TUD)
 D.C. Coleman (Elsevier Science Publishers)
 C.G. van der Laan
 J. Grootenhuis (CIRCE)
 N.A.F.M. Poppelier (Elsevier Science Publishers)
11. (werkgroep is vervallen)
12. **Beheerders handleiding/documentatie**
 J.L. Braams (PTT Research Neher Lab)
E.J. Evers (RUU; Geneeskunde) (coördinator)
13. **Nederlandstalige T_EX**
 J.L. Braams (PTT Research Neher Lab)
 V. Eijkhout (Univ. of Illinois)
 D. van Leeuwen (RUL)
 N.A.F.M. Poppelier (Elsevier Science Publishers)
14. **Communicatie**
J.L. Braams (PTT Research Neher Lab) (coördinator)
 V. Eijkhout (KUN)
 E.J. Evers (RUU; Geneeskunde)
 P. van Oostrum (RUU)
15. **T_EX 3.0**
 H.P.A. Mulders (KUB)
P. van Oostrum (RUU) (coördinator)
 E.J. Vens (RUG)

BIJLAGE C

Jaarverslag NTG 1990

Nederlandstalige T_EX Gebruikersgroep

Gerard van Nes

Maart 1991

Algemeen

In 1990 ging de NTG zijn 3^e jaar van bestaan in. Een succesvolle SGML & T_EX conferentie werd georganiseerd, in samenwerking met de SGML Users's Group Holland, cursussen werden aangeboden, een tweetal MAPS (Minutes & Appendices) verschenen, twee NTG bijeenkomsten vonden plaats, en er werd gewerkt aan de continuering van de NTG.

Het NTG bestuur

Het NTG bestuur bestond uit de volgende personen:

- C.G. van der Laan, voorzitter
- G.J.H. van Nes, secretaris
- J.L. Braams, penningmeester
- H.P.A. Mulders, bestuurslid
- T. de Klerk, bestuurslid

Er vonden in 1990 geen bestuurswisselingen plaats. Besloten werd om vanaf 1991 roulerend, twee bestuursleden (herkiesbaar) te laten aftreden.

Het NTG ledenbestand

Het jaar 1990 was het eerste jaar waarin contributie geheven werd. Voor die tijd waren in principe alle T_EX belangstellenden lid. Vanaf 1990 was voor individuele leden een contributie ingesteld van f. 75,- en voor instituutsliden f. 200,-.

Het uiteindelijk aantal leden dat aanbleef in 1990 was uitzonderlijk hoog. Van de 109 geregistreerde leden (T_EX belangstellenden) eind 1989, bleven uiteindelijk 71 lid in 1991. Daarnaast melden zich in de periode t/m december 1990 zich nog 46 leden aan waardoor het uiteindelijk ledental eind 1990 117 bedroeg. Hierbij inbegrepen 21 instituutsliden.

De volledige NTG ledenlijst werd gepubliceerd in de MAPS 90.2 en eerder in het jaar tevens verstuurd naar alle geregistreerde leden.

NTG bestuursvergaderingen

Op 16 januari, 5 oktober en 12 november vonden telefonische NTG bestuursvergaderingen plaats waarin naast de algemene gang van zaken binnen de NTG, de toestand binnen de Europese LUG's (Local User Groups) en TUG ter sprake kwam. Daarnaast werd aandacht besteed aan ondermeer ledenwerving en de NTG toekomststrategie.

De NTG bijeenkomsten

Er zijn in 1990 twee NTG bijeenkomsten georganiseerd welke gekenmerkt waren door een hoge mate van informatie uitwisseling:

1. Op 11 mei 1990 bij de Katholieke Universiteit te Nijmegen.
Aanwezig waren 42 leden.
2. Op 20 november 1990 bij Digital Equipment bv te Utrecht.
Aanwezig waren 38 leden.

Op deze bijeenkomsten werden ondermeer de volgende algemene lezingen gehouden:

- 'SGML en T_EX in een wetenschappelijke uitgeverij' door N.A.F.M. Poppelier,
- 'Communicatie allerlei' door J.L. Braams,
- 'SGML en T_EX' door C.G. van der Laan.

Van bovengenoemde twee vergaderingen verschenen uitgebreide verslagen met bijlagen (MAPS # 90.1 en 90.2). Zowel de kwaliteit als de kwantiteit van deze MAPS was in een sterke opgaande lijn.

De NTG dagen

Eind augustus/begin september werd in samenwerking met de SGML User's Group Holland, een SGML & T_EX conferentie georganiseerd. Aanwezig waren 90 deelnemers (inclusief sprekers), waaronder 5 Amerikanen, 2 Engelsen, 8 Duitsers en 1 Zwitser.

Op T_EX gebied vond een driedaagse cursus 'T_EX Intermediate' en 'T_EX Advanced' plaats.

Een verslag van deze bijeenkomst inclusief de volledige

tekst van de belangrijkste lezingen verscheen in de NTG MAPS # 90.2.

Werkbezoeken

Namens de NTG werden in 1990 de volgende internationale bijeenkomsten bezocht:

- GUTenberg '90
Van 14-18 mei werd door de voorzitter deelgenomen aan deze Franse T_EX Gebruikersbijeenkomst,
- EuroT_EX'90 te Cork
Aan deze Europese T_EX gebruikersbijeenkomst nam namens de NTG eveneens de voorzitter deel. De Board of Directors (BoD) bijeenkomst was inoverend. De Euro-top was interessant vanwege de kennismaking met de Oost-Europese afgevaardigden.

Van beide bezoeken is door de voorzitter uitgebreid verslag gedaan in de MAPS # 90.2.

De NTG werkgroepen

Een aantal werkgroepen waren in 1990 zeer actief hetgeen ondermeer resulteerde in het beschikbaarstellen c.q. distribueren van de uitstekende MS-DOS T_EX implementatie emT_EX, onderzoek en verspreiding van specifieke Nederlandstalige T_EX zaken, onderzoek en ondersteuning op het gebied van T_EX communicatie, verspreiding van informatie op het gebied van SGML & T_EX, en informatieverstrekking over T_EX versie 3.0.

In de beide MAPS van 1990 verschenen diverse bijdragen van de werkgroepen.

Nieuwe bestuursactiviteiten

Het bestuur heeft het beleid uitgezet de Nederlandse wiskundigen te winnen voor het gebruik van T_EX. Dit in samenwerking met het CWI en het Wiskundig Genootschap, naar analogie met de AMS in Amerika. Daarnaast heeft educatie en de PD PC-distributie haar speciale aandacht.

Diversen

- Goede contacten zijn er tussen de NTG en de andere Europese T_EX gebruikersgroepen. Besproken zijn samenwerkingsmogelijkheden onderling en tussen de LUG's en TUG zelf.
Daarnaast heeft de goede samenwerking met de Amerikaanse T_EX gebruikersgroep geleid tot het opnemen van de voorzitter van de NTG als een van de speciale Vice Presidents binnen de BoD en tot het helpen organiseren van EuroT_EX90, Cork.
- Verspreiding vond plaats van de laatste T_EX/L^AT_EX implementaties voor MS/PC-DOS, VAX en UNIX systemen.
- Van zowel de TEX-NL fileservers als de TEX-NL listserver werd in 1990 veel gebruik gemaakt. Het aantal abonnees van de listserver steeg in 1990 van 88 naar 126. Daarnaast werd via de NTG, diverse bestanden op de fileservers geplaatst.
- Door NTG leden werd een belangrijk aantal artikelen gepubliceerd in TUGboat, het internationale tijdschrift van de T_EX Users Group.
Eijkhout, Victor (1990): Unusual paragraph shapes. TUGboat 11#1, 51-53.
Poppelier, Nico & Johannes Braams (1990): A style option to adopt the standard L^AT_EX documentstyles to A4 paper. TUGboat 11#4, 98-102.
Eijkhout, Victor (1990): New books on T_EX. TUGboat 11#2, 210-211. TUGboat 11#4, 572.
Laan, Kees van der (1990): Typesetting bridge via T_EX. TUGboat, 11#2, 265-275.
Eijkhout, Victor (1990): An indentation scheme. TUGboat 11#4, 613-615.
Eijkhout, Victor (1990): A parskip scheme. TUGboat 11#4, 616-619.

Ook werd verslag gedaan van het wel-en-wee van de NTG zowel mondeling op de diverse BoD bijeenkomsten als schriftelijk in TUGboat 11#3, 446: NTG's second year.

Daarnaast werden door een viertal leden lezingen gegeven op de EuroT_EX90 conferentie in Cork/Ierland. Victor Eijkhout trad tevens als docent op voor de T_EX advanced cursus.

BIJLAGE D

Financiële verslag 1990

Nederlandstalige T_EX Gebruikersgroep

Johannes Braams

Maart 1991

1 Inleiding

In 1990 heeft de vereniging NTG voor het eerst in haar bestaan de beschikking gehad over financiële middelen. Dit betekent dat over 1990 een financiële verslag gemaakt moet worden. In een dergelijk verslag komen

een winst en verliesrekening en een begin- en eindbalans voor. In dit verslag ontbreekt de beginbalans, de vereniging had op 1 januari 1990 immers nog niet de beschikking over enigerlei financiële middelen. De overige onderdelen worden in het hiernavolgende kort besproken.

Table 1: De winst en verliesrekening over 1990

	Debet	Credit
Contributie		f 7,825.00
Sponsoring		
Rente		f 199.04
Evenementen		f 14,503.59
Administratie	f 325.00	
KvK en notaris	f 587.75	
Bestuurskosten	f 328.50	
Computer faciliteiten		
Mededelingen a/d leden	f 5,525.66	
Reisbijdragen	f 1,391.50	
Representatie	f 55.00	
Diversen Saldo	f 14,314.22	
	<u>f 22,527.63</u>	<u>f 22,527.63</u>

2 De winst en verliesrekening

In tabel 1 is de winst en verliesrekening over 1990 weergegeven. De diverse posten worden in de volgende paragrafen kort toegelicht.

2.1 Inkomsten

- **Contributies**

De vereniging heeft in het verslagjaar voor de eerste maal contributie gevraagd van haar leden. Aan het begin van het jaar telde de vereniging ruim honderd leden; in de begroting was rekening gehouden met de mogelijkheid dat ongeveer de helft van de geregistreerde leden zou afhaken op het moment dat contributie geheven zou worden. Gelukkig is gebleken dat het bestuur bij het opstellen van de begroting een te sombere kijk op de zaken heeft gehad. Ongeveer twee derde van de op 1 januari 1990 geregistreerde

leden is lid gebleven. Daarnaast is de groei van het aantal leden in 1990 doorgegegaan, waardoor aan het eind van 1990 117 geregistreerde en betalende leden aanwezig waren, waarvan 59 persoonlijke leden, één studentlid en 21 instituutleden met gezamenlijk 57 vertegenwoordigers.

- **Evenementen**

Hoewel de vereniging meer leden mocht registreren dan was verwacht is het grootste gedeelte van de inkomsten van de vereniging niet uit de contributies afkomstig. De oorzaak hiervan is gelegen in het feit dat de cursussen die bij de NTG-dagen in Groningen zijn gehouden een onverwacht hoog positief saldo hebben opgeleverd. Daarnaast is bij de post 'evenementen' ook het resultaat van de NTG-dagen in Utrecht 1989 opgenomen.

- **Rente**

Vanwege de hogere inkomsten dan verwacht heeft

de vereniging al in haar eerste financiële verslagjaar renteinkomsten kunnen genieten.

2.2 Uitgaven

- **Administratie**

Deze post bevat uitgaven voor het voeren van de administratie zoals portokosten, enveloppen, ordners, de kosten van de postbus.

- **KvK en notaris**

Deze post bevat de uitgaven voor het notariële laten passeren van de statuten en de inschrijving bij de kamer van koophandel in Groningen.

- **Bestuurskosten**

Deze post bestaat uit de kosten van telefonische bestuursvergaderingen.

- **Mededelingen aan de leden**

Deze post bevat de uitgaven gedaan voor het maken, reproduceren en verzenden van het 'verslag met bijlagen'.

- **Reisbijdragen**

De voorzitter heeft in 1990 twee maal een reis gemaakt waarvoor hij een bijdrage van de NTG heeft gevraagd. De eerste reis was naar Frankrijk om de NTG te vertegenwoordigen bij een bijeenkomst van GUTenberg. De tweede reis was naar de Europese T_EX bijeenkomst in Cork.

- **Representatie**

Het bestuur heeft de organisatiecommissie van de NTG

dagen in Utrecht een geschenk als dank aangeboden.

- **Saldo**

Deze post is geen uitgave, maar het bedrag dat de vereniging in 1990 heeft overgehouden. Uit deze post wordt duidelijk dat de vereniging een financieel succesvol jaar achter de rug heeft.

3 De balans

In tabel 2 is de balans per 31 december 1990 weergegeven. Uit de balans blijkt dat de vereniging een financieel gezond jaar achter de rug heeft. Ook is te zien dat een aantal leden nog geen contributie heeft betaald.

Het gaat hier om leden die op enigerlei wijze te kennen hebben gegeven dat zij lid willen blijven van de vereniging maar nog niet aan hun financiële verplichtingen voldaan hebben op 31 december 1991.

Daartegenover staat dat een aantal leden hun contributie voor het verenigingsjaar 1991 reeds vóór 31 december 1990 had betaald. In dit geval gaat het om leden die zich ná 1 september hebben aangemeld. Het bestuur heeft besloten in dergelijke gevallen de contributiebetaling te laten gelden voor de rest van het lopende verenigingsjaar én het volgende jaar.

Table 2: De balans per 31 december 1990

Eindbalans 1990	Aktiva	Passiva
Giro	f 14,114.22	
Contributies	f 1,050.00	f 850.00
Kapitaal		f 14,314.22
	f 15,164.22	f 15,164.22

4 Conclusie

De vereniging heeft een paar grote meevallers gehad in 1990, meer leden dan verwacht en een hogere opbrengst van de NTG-dagen dan verwacht. Het gevolg hiervan

is dat de vereniging nu een solide financiële basis heeft om de toekomst tegemoet te gaan. Het is een geruststellende gedachte dat eventuele tegenvallers, mochten die zich voordoen, opgevangen kunnen worden.

BIJLAGE E

Concept begroting van de Nederlandstalige T_EX Gebruikersgroep voor het jaar 1991

Hieronder vindt U de begroting voor 1991 van de Nederlandstalig T_EX gebruikersgroep. Een toelichting volgt na de tabel.

Inkomsten		Uitgaven	
Contributie	<i>f</i> 9100,00	Administratie	<i>f</i> 600,00
Sponsoring	"	Kamer van Koophandel	" 61,00
Saldo NTG-dagen '91	" 1200,00	Bijeenkomsten	"
Rente	" 400,00	Bestuurskosten	" 600,00
	"	Computerfaciliteiten	PM
		Nieuwsbrief/Verslagen	" 6000,00
		Reis bijdragen	" 3000,00
		Representatie	" 300,00
		Onvoorzien	" 139,00
	<hr/> <i>f</i> 10700,10		<hr/> <i>f</i> 10700,10

Toelichting

- **Inkomsten**

1. Contributie

De post contributie is gebaseerd op het aantal leden in januari 1991. Dat bedroeg:

21	insituten	$21 \times f 200,00$	<i>f</i> 4200,00
2	studenten	$2 \times f 50,00$	<i>f</i> 100,00
64	personen	$64 \times f 75,00$	<i>f</i> 4800,00
			<hr/> <i>f</i> 9100,00

2. Sponsoring

Er wordt geen sponsoring verwacht.

3. Saldo NTG-dagen

De NTG-dagen hebben in 1990 veel opgebracht, vooral dank zij de cursussen, we kunnen er niet vanuit gaan dat het dit jaar ook zo'n succes wordt.

4. Rente

De vereniging heeft in 1990 een behoorlijk kapitaal opgebouwd. Als dit niet nodig is om een tegenvaller op te vangen moet het mogelijk zijn behoorlijk wat rente te krijgen.

- **Uitgaven**

1. Administratie

Dit is bedoeld voor materiaal voor de secretaris en penningmeester. De hoogte is bepaald aan de hand van de hoogte van de uitkomst over 1990.

2. Kamer van Koophandel

Dit is een jaarlijks terugkerende inschrijving van

f 61,00.

3. Bestuurskosten

Hieronder vallen kosten als telefonische vergaderingen (*f* 600,00), vergoeding reiskosten voor een eventuele fysieke bijeenkomst etc.

4. Computerfaciliteiten

We maken gebruik van fileserver faciliteiten. Die worden op dit moment niet in rekening gebracht, maar dit kan in de toekomst wel eens veranderen. Vandaar dat dit als PM-post wordt opgevoerd.

5. Nieuwsbrief/Verslagen

Het kopiëren en verspreiden van de verslagen van de bijeenkomsten. De kosten bedragen ongeveer *f* 20,00 per exemplaar.

6. Reisbijdragen

Het is de bedoeling dat de vereniging bijdraagt in de kosten van het bijwonen van buitenlandse bijeenkomsten die met T_EX te maken hebben. Als tegenprestatie wordt een verslag van de bijeenkomst verwacht, ter publicatie binnen de vereniging.

7. Representatie

Als bestuursleden van zusterverenigingen bij onze bijeenkomst uitgenodigd worden wordt een tegevoetkomst in de kosten gegeven.

8. Onvoorzien

Spreekt voor zich.

BIJLAGE F**TeX-NL subscription**

28 maart 1991

TeX-NL is de Nederlandstalige TeX-informatie distributielijst (ook wel discussielijst genoemd). Het adres is:

TEX-NL@HEARN

Men kan zich op deze TeX-NL discussielijst abonneren (TEX-NL mails ontvangen en versturen) via de volgende VAX/VMS commando's (of analoge commando's voor andere computer systemen):

```
$
$ SEND LISTSERV@HEARN
  > SUBSCRIBE TEX-NL  your_name
$
```

Een lijst van deelnemers is te verkrijgen via de commando's:

```
$
$ SEND LISTSERV@HEARN
  > REVIEW TEX-NL
$
```

Met als resultaat:

```
*
*   TEX-NL
*
*   Review=      Public
*   Subscription= Open
*   Send=        Public
*   Notify=      Yes
*   Reply-to=    List,Ignore
*   Files=       Yes
*   Validate=    Store only
*   Errors-To=   Owners
*   X-Tags=      Comment
*   Stats=       None,Private
*   Confidential= No
*
*   owner= Quiet:,U070007@HNYKUN11 (Niek Cox)
*   owner= Quiet:,BRAAMS@HLSDDL5 (Johannes Braams)
*   owner= EVERS@HUTRUU53 (Evert Jan Evers)
*
*
*   pfuetz@AGD.FHG.DE           Matthias Pfuetzner, ZGDV Darmstadt
*   STOKHOF@ALF.LET.UVA.NL      Martin Stokhof
*   VDBERG@ALF.LET.UVA.NL      Martin H. vdBERG
*   KROPVELD@AMC.UVA.NL        Dani"el Kropveld
*   CI@ANALYSIS.RUG.AC.BE      Chris Impens
*   mattes@AZU.INFORMATIK.UNI-STUTTGART.DE Eberhard Mattes
*   GEOMETRY@BGERUG51          FRANK DE CLERCK
*   LAAAA18@BLEKUL11           Erik van Eynde
*   LAAAA41@BLEKUL11           Marleen Vander Straeten
*   hans@CS.KUN.NL             Hans Meijer
*   phons@CS.KUN.NL            Phons Bloemen
*   piet@CS.RUU.NL             Piet van Oostrum
*   vansoest@CS.UTWENTE.NL     Dick van Soest
*   eijkhout@CSR.D.UIUC.EDU     Victor Eijkhout
*   GPARTOS@DGIHRZ01           G. Partosch, HRZ Univ. Giessen, F.R.G.
*   GPTEX@DGIHRZ01             TeX-Install., HRZ Univ. Giessen, F.R.G.
*   DANTE@DHDURZ1             DANTE e.V.
*   nust@DUTENTB.ET.TUDELFT.NL Jan H Nusteling
*   abi@DUTIAA.TUDELFT.NL      Ton Biegstraaten
```

wim@DUTIOSA.TUDELFT.NL	Wim Penninx
wiorst5@DUTIWS.TUDELFT.NL	Bert van Zomeren
osxul000@DUTIWX.TUDELFT.NL	Hildo Biersma
mkmfhuy@DUTREX.TUDELFT.NL	Tom Huijgen
wbtrvos@DUTREX.TUDELFT.NL	Ron v. Ostayen
kees@DUTTWTA.TUDELFT.NL	C.L. Koster
robk@DUTTWTA.TUDELFT.NL	Rob Kuyper
jaap@DUTW9.TUDELFT.NL	Jaap van der Zanden
martien@DUTW9.TUDELFT.NL	Martien Hulsen
hogenbirk@ECN.NL	Alfred Hogenbirk
winnink@ECN.NL	J.J. Winnink
N.POPPELIER@ELSEVIER.NL	"Nico Poppelier"
COMBEE@ET.TUDELFT.NL	leendert combee
VAN-ZWYN@FRECP12	C. van Zwynsvoorde
HELLINGS@HASAMC51	JAN HELLINGS
A401WIJZ@HASARA11	Maurits Wijzenbeek
A403KOX@HASARA11	A.J. Kox
A410SAKE@HASARA11	Sake J. Hogeveen
A471HANS@HASARA11	hans van der meer
A9530020@HASARA11	repke de vries
SOND0016@HASARA11	R Veldhuyzen van Zanten
EMMEN@HASARA5	Ad Emmen
dee%svcentlv@HDEDH1	Dick Dee
RCDILAA@HDETUD1	Hans van der Laan
WBAHKUI@HDETUD1	Gerard Kuiken
WIORA03@HDETUD1	Netty Zuidervaart
TNLTSMC@HDETUD1.TUDELFT.NL	John VanderKoiijk
JIAN@HDETUD11	Jian LUO
DENHAAN@HDETUD5	Jack den Haan
WBWEAHA@HDETUD51	J.B.W. HOEBEEK
RCRONH@HEITUE5	Ron Helwig
RCGBBATG@HEITUE51	"Tonnie Geraets
ALDHAHIR@HENUT5	Alaaddin Al-Dhahir
TWPOLDER@HENUT5	Jan Willem Polderman
henk@HGATENL.HOBBY.NL	Henk Dijkstra
BOSVELD@HGRRUG5	"Gerard Bosveld"
CGL@HGRRUG5	CG VAN DER LAAN
DRUNEN@HGRRUG5	Rudi van Drunen
FRAMBACH@HGRRUG5	ERIK FRAMBACH
KONING@HGRRUG5	RUUD H. KONING
TAJ@HGRRUG5	Theo Jurriens
LEEGTE@HGRRUG51	Henk Leegte
SPORREL@HGRRUG51	frans sporrel
STOOP@HGRRUG51	"Paul Stoop"
KANABY@HHEOUH51	Abdy Jooya
PUXHHO@HHEOUH51	Bert Houben
LETTVA@HLERUL2	Andrea de Leeuw van Weenen
WITWAA@HLERUL2	"J.A. van de Griend"
HERMANS@HLERUL5	FRANK HERMANS
FTHKOPER@HLERUL52	GER KOPER
BORSBOOM@HLERUL53	G.J.J.M. Borsboom
VDSCHOOT@HLERUL53	Jan Vanderschoot
DAVID@HLERUL59	David van Leeuwen
BRAAMS@HLSDNL5	Johannes Braams
MFAGKCHR@HMARL5	CHRIS EVELO
U070007@HNYKUN11	NIEK COX
U070040@HNYKUN11	Patrick Wever
U212307@HNYKUN11	Peter Bronts
U216002@HNYKUN11	Paul Wackers
U250005@HNYKUN11	Peter-Arno Coppen
U251006@HNYKUN11	Hans Stoks
U253002@HNYKUN11	Constant Cuypers
U279102@HNYKUN11	Theo van den Heuvel
U439019@HNYKUN11	Ton de Haan
U605005@HNYKUN11	Willem Jan Karman
U605008@HNYKUN11	Rik Fleuren
U641012@HNYKUN11	Rini van Doorn
MBFYS_NU@HNYKUN51	"Hong Zhou"
CAOS@HNYKUN52	HENS BORKENT
BISON@HNYKUN53	"Pieter Bison"
SYLVIA@HNYMPI51	"Sylvia Aal"
GAVIN@HNYMPI52	"Gavin Burnage"
VANNES@HPEENR51	Gerard van Nes
VDENDE@HROEUR5	"Jan van der Ende"

```

HUYGEN@HROEUR51
GENT@HTIKUB5
SPANJERS@HTIKUB5
S172HMUL@HTIKUB5
EVERS@HUTRUU53
erikjan@ICCE.RUG.NL
HAAN@IRIVAX.TUDELFT.NL
JENSE@ITI.TNO.NL
SURF114@KUB.NL
WATSON@LCC.EDU
andre@MAESTRO.HTSA.AHA.NL
G.D.Draaijer@MARIN.NL
NSEV@MARIN.NL
R.H.M.Huijsmans@MARIN.NL
bnb@MATH.AMS.COM
koole@MATH.LEIDENUNIV.NL
demeijer@MATH.RUU.NL
hvdberg@MATH.UTWENTE.NL
soos@MATH.UTWENTE.NL
aerts@MEDIA01.UUCP
paulks@NIKHEFK.NIKHEF.NL
MBR@OCE.NL
hd@PHILTIS.CFT.PHILIPS.NL
AK@PLWRTU11
d5@PMS.UIA.AC.BE
BRUINSMA@RUG.NL
PZF5HZ@RUIPCLE
BILT@RUUMTC.TCU.RUU.NL
nspit@RUUNSA.FYS.RUU.NL
spit@RUUNSA.FYS.RUU.NL
RWBEST@SARA.NL
petervc@SCI.KUN.NL
NETNEWS@TREARN
STELP@TUDGV1.TUDELFT.NL
HUISMAN@TUDW03.TUDELFT.NL
TH788310@TWNCTU01
rcpt@URC.TUE.NL
KALPAKLI@UWAV1.U.WASHINGTON.EDU
KNAPPEN@VKPMZD.PHYSIK.UNI-MAINZ.DE
*
* Total number of "concealed" subscribers:      2
* Total number of users subscribed to the list: 132 (non-"concealed" only)
* Total number of local node users on the list:  0 (non-"concealed" only)
*

```

Opmerkingen:

- Verzocht wordt om de TEX-NL listserver niet te gebruiken voor het versturen van grote bestanden (programma's) indien van het alternatief: **de TEX-NL fileserver** (zie bijlage G), gebruik gemaakt kan worden.
- Daar ook enkele buitenlanders meeluisteren, wordt men verzocht de 'subject' van de mail in het Engels op te geven.
- De TEX-NL listserver is bij uitstek geschikt voor een verzoek voor ondersteuning bij een $\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ /driver probleem, voor vragen over beschikbaarheid van bepaalde software modulen, voor aankondigingen van bijeenkomsten en/of cursussen, voor het attenderen op bepaalde publicaties, voor het attenderen op bepaalde produkten, voor een mededeling die ook voor een grotere groep interessant is, etcetc..

BIJLAGE G**NTG fileserver faciliteiten**

28 maart 1991

Sinds mei 1989 heeft NTG de TEX-NL fileserver. Voor leden interessante files worden daarbij centraal beschikbaar gesteld.

Men kan files van deze fileserver betrekken via de volgende VAX/VMS commando's (of analoge commando's voor andere computer systemen):

```
$
$ SEND LISTSERV@HEARN
  > GET filename filetype
$
```

Waarbij de mogelijke *filenames* en *filetypes* in de hieronder getoonde listing zijn opgenomen.

De lijst van alle aanwezige files is te verkrijgen via de commando's:

```
$
$ SEND LISTSERV@HEARN
  > GET TEX-NL FILELIST
$
```

Met als resultaat:

```
* TEX-NL FILELIST for LISTSERV@HEARN.
*
* TeX-NL Filelist
*
* Contains
*   -- general TeX stuff (implementations for micros, graphical
*   shells, printer drivers)
*   -- specifically Dutch stuff (styles and options, hyphenation
*   patterns)
*   -- Dutch TeX Users Group (NTG) stuff
*
* *****
*
* This file lists the programs that are stored on LISTSERV and can be
* retrieved by network users.
*
* If an entry shows nrecs=0 the file is not available.
*
* This filelist may be sorted in columns 47 to 63 to get a list of
* files in the order of their updates. Sorting in descending order
* shows the most recently updated files at the top.
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
*
* The GET/PUT authorization codes shown with each file entry describe
* who is authorized to GET or PUT the file:
*
*   ALL = Everybody
*   N/A = Not Applicable
*   LCL = Local users, as defined at installation time
*   PRV = Private, ie list members
*   OWN = List owners
*   NAD = Node Administrators, ie official BITNET/EARN contacts
*   CTL = LISTEARN Controllers (Also called "Postmasters")
*
*::  NTG = 'BRAAMS@HLSDDL5', /* Johannes Braams */
*::  'BRAAMS@HLSDDL50', /* Johannes Braams */
*::  'BRAAMS@HLSDDL51', /* Johannes Braams */
*::  'BRAAMS@HLSDDL52', /* Johannes Braams */
*::  'U641000@HNYKUN11', /* Victor Eijkhout */
*::  'U641001@HNYKUN11', /* Victor Eijkhout */
*::  'EIJKHOUT@CSR.D.UU' /* Victor Eijkhout */
```

```

*:          'EVERS@HUTRUU53'      /* Evert Jan Evers          */
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
*
*****
*
* Dutch hyphenation patterns
*
* Hyphen1 TeX : shortened Celex-list, all lines with a 5 in them removed,
*               in order to be able to load it when you can't stretch
*               the 'triesize'
* Hyphen2 TeX : long and powerful (author: Celex, Nijmegen)
*               Note that this requires stretching the 'triesize'
*               of both TeX and IniTeX!
* UShyphen ADD: extra patterns to handle the Tugboat exception log
*               (author: Gerard Kuiken)
*
*****
*
*               rec                last - change
* filename filetype  GET PUT -fm lrecl nrecl  date      time      File description
* -----
* HYPHEN1  TEX        ALL NTG V      80   6122 90/12/20 13:15:31
* HYPHEN2  TEX        ALL NTG V      72   7949 89/05/19 20:21:37
* USHYPHEN ADD      ALL NTG V      73    378 90/05/14 13:20:11
*
*****
*
* Options for Dutch
*
* A4 STY    : A4-paper width and height
*            by Nico Poppelier and Johannes Braams (historical order)
*            Note that this is not the A4 option of John Pavel.
* A4 TeX and A4 DOC: Accompanying documentation for A4.STY
* Dutch old : Redefines captions and does other useful things for
*            all standard document styles. (author: Johannes Braams)
*            This is really an international option.
*            This file has been superseded by the dutch.sty in the
*            BABEL system (See further on)
* German STY: The style on which 'Dutch' was based. The two are
*            compatible. (author: Hubert Partl)
* Sober STY : Reduces section headings and white spaces a bit;
*            this is only repair for the standard styles. The official
*            NTG styles (below) can do without. (author: Nico Poppelier)
*
*****
*
*               rec                last - change
* filename filetype  GET PUT -fm lrecl nrecl  date      time      File description
* -----
* A4        STY        ALL NTG V      80    135 91/02/13 10:50:05
* A4        DOC        ALL NTG V      80    511 91/02/13 13:58:32
* A4        TEX        ALL NTG V      80     39 91/02/13 10:49:00
* DUTCH     OLD        ALL NTG V      80    397 90/12/20 18:45:23
* GERMAN    STY        ALL NTG V      76    364 89/06/24 16:08:38
* SOBER     STY        ALL NTG V      77    147 89/06/24 16:06:16
*
*****
*
* The BABEL system
*
* This is a pre-release of the BABEL system. Therefore only a few
* language specific files are provided. When an article about the
* system has appeared in TUGboat all available files will be placed
* here.
*
* See the file BABEL README for further instructions
*
*****
*
* BABEL  README  ALL NTG V      80     40 90/12/20 18:37:14
* BABEL  DOC     ALL NTG V      80    915 90/12/20 18:44:49
* BABEL  STY     ALL NTG V      80    245 90/12/20 18:37:37
* DUTCH  DOC     ALL NTG V      80    416 90/12/20 18:45:53
* DUTCH  STY     ALL NTG V      80    129 90/12/20 18:38:00
* ENGLISH DOC    ALL NTG V      80    202 90/12/20 18:38:26
* ENGLISH STY    ALL NTG V      80     97 90/12/20 18:40:02

```

```

GERMANB DOC      ALL NTG V      80   594 90/12/20 18:46:37
GERMANB STY      ALL NTG V      80   223 90/12/20 18:40:29
LATEXHAX DOC     ALL NTG V      80    67 90/12/20 18:43:09
LATEXHAX STY     ALL NTG V      80    52 90/12/20 18:43:37
*****
*
* Dutch styles (author: Victor Eijkhout)
*
* Completely compatible to 'article' and 'report', but improved layout;
* these styles have as default language English,
* for Dutch or German add corresponding style options
*
* Artikel1 doc : Article-compatible, tight look, documented (somewhat)
* Artikel1 sty : without documentation
* Artikel2 doc : Article-compatible, heavily indented; quite something else
* Artikel2 sty : without documentation
* Artikel3 doc : Article-compatible; zero parindent, positive parskip;
*               otherwise similar to Artikel1
* Artikel3 sty : without documentation
* Rapport1 doc : Report-compatible; looks like Artikel1
* Rapport1 sty : without documentation
* Rapport2 doc : will probably not come into being.
* Rapport2 sty : without documentation
* Rapport3 doc : Report-compatible; looks like Artikel3
* Rapport3 sty : without documentation
* Boek doc     : Book-compatible; artikel1 layout
* Boek sty     : without documentation
*
* Options for the Dutch styles
*
* Ntg10 doc    : 10point option for all styles
* Ntg10 sty    : without documentation
* Ntg11 doc    : 11point option for all styles
* Ntg11 sty    : without documentation
* Ntg12 doc    : 12point option for all styles
* Ntg12 sty    : without documentation
* Voorwerk doc : Replaces Titlepage.STY for report styles
* Voorwekr sty : without documentation
*
* NTGstyle UUE : All in one buy; UUencoded ZOO archive (see below
*               for ZOO)
*****
*
*               rec                last - change
* filename filetype GET PUT -fm lrecl nrecl  date      time      File description
* -----
ARTIKEL1 DOC     ALL NTG V      80  1204 91/02/26 21:26:37
ARTIKEL1 STY     ALL NTG V      80   618 91/02/26 21:28:09
ARTIKEL2 DOC     ALL NTG .        .    0 .....
ARTIKEL2 STY     ALL NTG V      80   570 90/12/20 12:15:30
ARTIKEL3 DOC     ALL NTG .        .    0 .....
ARTIKEL3 STY     ALL NTG V      80   648 91/03/22 19:01:04
RAPPORT1 DOC     ALL NTG .        .    0 .....
RAPPORT1 STY     ALL NTG V      80   654 91/03/22 19:03:50
RAPPORT2 DOC     ALL NTG .        .    0 .....
RAPPORT3 DOC     ALL NTG .        .    0 .....
RAPPORT3 STY     ALL NTG V      80   660 91/03/22 19:03:01
BOEK DOC        ALL NTG .        .    0 .....
BOEK STY        ALL NTG V      80   682 91/02/21 11:24:43
NTG10 DOC       ALL NTG V      80   193 91/02/26 12:12:44
NTG10 STY       ALL NTG V      80   165 91/02/26 21:13:36
NTG11 DOC       ALL NTG V      80   197 91/02/26 12:14:29
NTG11 STY       ALL NTG V      80   168 91/02/26 21:14:13
NTG12 DOC       ALL NTG V      80   196 91/02/26 12:15:32
NTG12 STY       ALL NTG V      80   169 91/02/26 21:16:49
VOORWERK DOC    ALL NTG .        .    0 .....
VOORWERK STY    ALL NTG V      69    58 90/04/17 12:38:26
NTGSTYLE UUE    ALL NTG V      80  5238 91/03/15 12:04:21
*****
*
* The letter style according to Dutch NEN norms (by Victor Eijkhout)
*
* BRIEF STY      : The style file
* BRIEF TeX     : An example letter

```

```

* BRIEFDOC TeX : Explanation of the options of the letter style
*
*****
*                               rec                last - change
* filename filetype  GET PUT -fm lrecl nrecl  date      time      File description
* -----
* BRIEF      STY      ALL NTG V      80   678 91/03/26 22:39:18
* BRIEF      TEX      ALL NTG V      72   198 89/10/24 16:48:49
* BRIEFDOC   TEX      ALL NTG V      80   283 90/11/02 14:02:00
*
*****
*
* The latest in TeXnology
*
* LATEXSRC UUE : (not quite) Latest versions of all LaTeX materials;
*               UUencoded ZOO archive
* ASCII TeX   : ASCII table (author: Victor Eijkhout)
* TUGBOAT COM : Common commands for Tugboat styles
* TUGBOAT STY : Plain TeX style for Tugboat article
* LTUGBOAT STY : LaTeX style for Tugboat articles
*
* BTXMAC.TEX  : BibTeX 0.99c macros for use with plain TeX.
*               The file specifies that is meant for TeX 3.0 or later
*
* MULTICOL    : The multicolumn package written by Frank Mittelbach and
*               Rainer Schoepf, as published in TUGboat.
*               The package includes DOC.STY. The package consists of two
*               files, MULTICOL README and MULTICOL ARCUUE. These should
*               distributed together.
*
* CHANGEBARS  : Michael Fine's changebar.sty, modified for use with plain
*               TeX as well as with LaTeX. Also modified to support DVIToPS
*               \specials as well as DVI2LN3 \specials
*
*****
*                               rec                last - change
* filename filetype  GET PUT -fm lrecl nrecl  date      time      File description
* -----
* LATEXSRC UUE      ALL NTG V      80 10763 91/02/20 13:51:06
* ASCII   TEX      ALL NTG V      82   194 89/10/20 20:06:20
* TUGBOAT COM      ALL NTG V      80   797 90/08/28 18:44:06
* TUGBOAT STY      ALL NTG V      80  2278 90/08/28 18:50:07
* LTUGBOAT STY     ALL NTG V      80   444 90/09/07 11:07:53
* BTXMAC   TEX     ALL NTG V      80   624 90/08/15 16:59:21
* MULTICOL README  ALL NTG V      80   100 90/10/15 11:05:35
* MULTICOL ARCUUE  ALL NTG V      80  1293 90/10/15 11:26:14
* CHNGBARS STY     ALL NTG V      80   828 91/03/15 11:13:28
*
*****
*
* Pleasant reading material about TeX and its uses
*
* NTGstyle TeX : Manual for the Dutch LaTeX styles
* Layout TeX   : Article about documentstyle development in LaTeX
*               Intended as supplement to chapter 5 LaTeX book
* Layout2 TeX  : Goes with previous; in German (Hubert Partl)
* Refman STY   : Needed for previous two
* Bridge TeX   : About setting bridge games in LaTeX (Kees van der Laan)
* Artdoc TeX   : The history of the 'Artikel' styles; almost a
*               manual for document style development; in Dutch
* Rapdoc TeX   : The same for the 'Rapport' styles (Victor Eijkhout)
* Gentle TeX   : A Gentle Introduction to TeX (Michael Doob)
*
*****
*                               rec                last - change
* filename filetype  GET PUT -fm lrecl nrecl  date      time      File description
* -----
* NTGSTYLE TEX     ALL NTG V      72   122 89/09/04 12:20:21
* LAYOUT   TEX     ALL NTG V      79  1090 89/06/26 12:12:34
* LAYOUT2  TEX     ALL NTG V      80  1011 89/06/26 12:03:15
* REFMAN   STY     ALL NTG V      79   492 90/03/26 18:12:17
* BRIDGE   TEX     ALL NTG V      75   415 89/06/26 11:54:36
* ARTDOC   TEX     ALL NTG V      71   708 89/09/04 12:21:36
* RAPDOC   TEX     ALL NTG V      75   735 89/09/04 12:22:13

```


GENTLE TEX ALL NTG V 79 5341 90/01/09 13:56:01

*
 * Nederlandstalige TeX Gebruikersgroep (Dutch TeX Users Group)
 *
 * Notuul1 TeX : Vergadering 23 juni 1988
 * Notuul2 TeX : Vergadering 24 november 1988
 * Notuul3 TeX : Vergadering 11 mei 1989 (3 bestanden: notuul3a,b,c)
 * TeXdag89 TeX : Verslag eerste Nederlandse TeXdagen 29/30 juni 1989
 *
 * Statuten TeX : De statuten van de vereniging NTG
 * Statuten sty : bijbehorende document stijl-optie
 *

* filename	* filetype	* rec			* last - change		* File description		
		GET	PUT	-fm	lrecl	nrecs		date	time
NOTUUL1	TEX	ALL	NTG	V	80	1043	89/06/26	11:50:54	
NOTUUL2	TEX	ALL	NTG	V	80	1457	89/06/26	11:52:06	
NOTUUL3A	TEX	ALL	NTG	V	80	108	89/10/30	10:12:47	
NOTUUL3B	TEX	ALL	NTG	V	80	1941	89/10/30	10:13:27	
NOTUUL3C	TEX	ALL	NTG	V	80	2634	89/10/30	10:14:05	
TEXDAG89	TEX	ALL	NTG	V	73	274	89/12/08	13:04:52	
STATUTEN	TEX	ALL	NTG	V	80	532	91/03/04	20:55:17	
STATUTEN	STY	ALL	NTG	V	80	94	91/03/04	14:14:21	

*
 * TeX for micros
 *
 * STZOO UUE : UUencoded ARC archive with ZOO for the Atari ST
 * MSZOO UUE : zoo.exe for MS-DOS, UUencoded
 * MSFIZ UUE : fiz.exe for MS-DOS, UUencoded
 * MSSUP201 UUE : MS-DOS support for zoo, ZOO archive, UUencoded
 * Z201SRC1 UUE : Sources of zoo, part 1, ZOO archive, UUencoded
 * Z201SRC2 UUE : Sources of zoo, part 2, ZOO archive, UUencoded
 * TEXSHELL UUE : TeX environment for the Atari ST, ZOOed
 * WP2LATEX UUE : WordPerfect to LaTeX translator, ZOOed
 *

* filename	* filetype	* rec			* last - change		* File description		
		GET	PUT	-fm	lrecl	nrecs		date	time
STZOO	UUE	ALL	NTG	F	61	2181	89/12/14	12:33:31	
MSZOO	UUE	ALL	NTG	V	61	946	90/03/18	20:53:06	
MSFIZ	UUE	ALL	NTG	V	61	297	90/03/18	20:54:11	
MSSUP201	UUE	ALL	NTG	V	61	621	90/03/18	20:55:29	
Z201SRC1	UUE	ALL	NTG	V	61	1975	90/03/18	20:58:36	
Z201SRC2	UUE	ALL	NTG	V	61	2046	90/03/18	20:59:21	
TEXSHELL	UUE	ALL	NTG	V	62	594	90/01/03	16:12:11	
WP2LATEX	UUE	ALL	NTG	V	80	1229	90/03/12	15:58:03	

*
 * METAFONT sources
 *
 * AMSREAD.ME : A few notes about the contents of AMSFONTS.UUE
 * AMSFONTS.UUE : The AMS font collection UUencoded ZOO archive
 * split in ten pieces of app. 100kByte
 *

* filename	* filetype	* rec			* last - change		* File description		
		GET	PUT	-fm	lrecl	nrecs		date	time
AMSREAD	ME	ALL	NTG	V	74	45	90/08/02	14:18:33	
AMSFONTS	UU1	ALL	NTG	F	80	1644	90/08/02	15:50:09	
AMSFONTS	UU2	ALL	NTG	F	80	1643	90/08/02	15:58:11	
AMSFONTS	UU3	ALL	NTG	F	80	1643	90/08/02	16:07:46	
AMSFONTS	UU4	ALL	NTG	F	80	1643	90/08/02	16:41:15	
AMSFONTS	UU5	ALL	NTG	F	80	1643	90/08/02	16:44:37	
AMSFONTS	UU6	ALL	NTG	F	80	1643	90/08/02	16:47:16	
AMSFONTS	UU7	ALL	NTG	F	80	1643	90/08/02	16:49:31	
AMSFONTS	UU8	ALL	NTG	F	80	1643	90/08/02	16:51:56	

```
AMSFONTS UU9      ALL NTG F      80 1643 90/08/02 16:53:58
AMSFONTS UUA      ALL NTG F      80 1659 90/08/02 16:55:44
```

Het adres van de Duitse fileserver (Heidelberg) is:

LISTSERV@DHDURZ1

Voor NTG leden die niet op een netwerk zijn aangesloten, kunnen de meeste files via diskettes verkregen worden. Nadere informatie hierover bij Gerard van Nes.

BIJLAGE H

Van de voorzitter

T_EX Nationaal en Internationaal

Oftewel: wat gebeurt er allemaal?

Maart 1991

NTG

Het 3^{de} NTG jaar is globaal een voortzetting van de voorgaande jaren. Goed nieuws is dat de WG PCs actief geworden is. Een belangrijk aandachtspunt is het handhaven van de goede sfeer.

Op het bestuurlijke vlak wordt een verkiezingsproces op gang gebracht, met als doel een democratische vereniging te creëren. De contacten met de zusterverenigingen worden voortgezet, c.q. uitgebreid. Dit uit zich o.a. in de EuroT_EXs. Aan de ontwikkeling van TUG naar een Internationale TUG wordt meegewerkt.

Een belangrijk beleidspunt is de actieve benadering van 'wetenschappers,' te beginnen met de wiskundigen, om het gebruik van T_EX als documentpreparatie gereedschap te bevorderen. Wij hopen de vraagstelling

'Hoe kan het gebruik van T_EX ,door wiskundigen, bevorderd worden?'

in overleg met het CWI en het Wiskundig Genootschap inhoud te geven. Wij gaan ervan uit dat WordPerfect in de academische wereld het meest gebruikt wordt, maar dat T_EX etc. voor complexe kopij, interessant en effectief kan zijn. Daarnaast zijn er de appelerende DTP-pakketten. SGML is geen concurrent. SGML heeft T_EX als formatter nodig.

Concreet voor de leden wordt er gewerkt aan een Info-pakket (aspirant leden), en een Welkomstpakket (nieuwe leden). De info-verspreiding via de Minutes+Appendices, verloopt boven verwachting. Ook de geplande presentaties op 2 Mei, zijn fantastisch. T_EX als gereedschap, in productie. Daar gaat het toch eigenlijk om.

Ook wordt gedacht aan het professioneler opzetten van de logistiek rondom distributie van PD PC T_EX etc. programmatuur. Ook wordt er ruime aandacht geschonken aan het educatie proces. Enne, . . . de NTG-dagen komen uit de lengte of de breedte. Het valt en staat allemaal

met de inzet van de leden.

TUG

Na een benauwende stilte voor de storm is de zon plotseling doorgebroken. Halleluja, iedereen zingt weer, en flaneert op de T_EX scene als nooit tevoren. TUG is alive! De BoD slinkt af —nooit slecht— naar 15 (verkozen) leden, zonder aparte EC of FC. De positie van de afgevaardigden uit zustergroeperingen —the special vice-presidents— zweeft nog. But who cares? Het bruist. Enthousiasme alom. De president zal in het vervolg via schriftelijke stemming onder alle leden verkozen worden. Hopelijk toekomstmuziek, cause he is a jolly good fellow.¹

Er zijn diverse commissies benoemd.² Er is een PD PC commissaris benoemd (John Radel), en enige commissies die zich met de begroting³, office zaken en operationele procedures bezig houden.

Eindelijk is de Long Range Planning committee⁴ ingesteld. De sprong voorwaarts is de Publicatie commissie⁴, die het hele publicatie en distributie gebeuren van TUG moet herbezien. Christina Thiele zat op hete kolen en wierp zichzelf in de strijd met het idee om voor de volgende bijeenkomst (AGM), TUG91, 15–18 juli, in Dedham, een proto-nieuwsbrief verspreid te hebben. Deze nieuwsbrief, DE 'root' nieuwsbrief van de internationale T_EX etc. gebruikersgroepen, wordt geacht te verschijnen in verschillende edities. De LUGs zorgen zelf voor de aanpassingen —m.b.t. taal en/of inhoud— en de verspreiding. Bravo!

Op de achtergrond rommelt er nog wat met de Knuth Scholarship commissie⁴, maar dit verbleekt onder de zon.⁵ De educatie commissie is tussen het wal en het

¹Rick Furuta, David Ness, en Cal Jackson hebben hun functie ter beschikking gesteld. Ook hebben wij geen Europese coordinator meer, whatever that meant.

²Het geheel kan binnen 30 dagen teruggedraaid worden als de afwezige meerderheid het er niet mee eens is.

³Er is geen tekort meer!

⁴Met mijn persoontje on it.

⁵Barbara Beeton pakt dit echter voortvarend op, en ik laat haar niet in de kou staan, zodat de zaak rond Juli aardig op poten zal staan. Geen toekenning van de Scholarship-award dit jaar echter, wegens verlapsing van de aankondiging, en eigenlijk het hele ad hoc gebeuren in deze.

schip gevallen. De professionalisering daar komt ongetwijfeld, in due time. Een belangrijk aandachtspunt. Wij werken eraan.

ukTeXug.

Baskerville is verschenen, de nieuwsbrief van onze burenen (zie leestafel). T_EXline blijft. Verder zijn de Britten behoorlijk actief: een aardig jaarprogramma, met in afgelopen Februari de L^AT_EX boost. Is het L^AT_EX3.0-project verder gebracht?

GUTenberg.

De Fransen nemen wederom het Europese voortouw. Zij organiseren Euro T_EX91, 23–26 sept, in Parijs. Malcolm

Clark organiseert de cursussen. Ah Paris, c'est la vie.

En verder . . .

DANTE wordt steeds dikker, ITALIC staat nog tussen de coulissen⁶, NORDIC neuriert vrolijk verder, de Oost-Europeanen zijn muisstil. Japan? De grote coup? . . . en wij? Wij gaan gewoon door met dijken bouwen, vatten de koe bij de horens:

Wiskundig Nederland,
Here we come!

⁶Peter Flynn vertelde mij dat zij nu officieel als een LUG bestaan. Hij is echter (nog?) geen Vice President van de BoD, whatever that means.

BIJLAGE I**Working Group 1: Education**

March 1991

1 Task

As a complementary note to the earlier report on education, see appendix to the MAPS of the 4th meeting, the following WG task description.

- To advise NTG about educational matters.
- To maintain a list of what courses are offered by whom, when and under what conditions.
- To participate in the international discussion about T_EX related courses, e.g. the standardization of the various modules.
- To stimulate the development of standard courseware.
- To stimulate the development of tools in order to create and maintain courseware.

2 What courses?

The WG strongly advises NTG to offer

- L^AT_EX (beginners, advanced and special topics(styles); L^AT_EX for personnel),
- T_EX (beginners, intermediate, advanced, and special topics, e.g. T_EXing math),
- Metafont (logo design, font design), along with the ‘NTG-days.’

Up till now, since the 4th NTG meeting, only Kees van der Laan submitted an announcement of courses (see Appendix); no modifications have been received. A request for a course for personnel has been received. This was taken care of by Piet Tutelaers; TUE absorbed those TUT people in their classes.¹

At the moment a course about ‘literate programming with Web,’ or Postscript is considered too early, c.q. not the issue of NTG.

Course modules

No follow-up of the Bart Childs discussion has taken place yet. At Cork the education issues were put on the agenda, but postponed. At the extra March BoD meeting it was submitted, but, hela, . . . too late. Kees will rise the issue again at Dedham, July 1991.

3 Teachers**3.1 International**

- **Amy Hendrickson** (T_EX and Postscript),
- **David Salomon** (Output routines (see his tutorials TUGboat 11# 1, 2, 4.)),
- **Doug Henderson**(Metafont),
- **Malcolm Clark, Chris Rowly**, (L^AT_EX beginners and advanced),

3.2 Local

See ‘maintained list,’ and Appendix.

3.3 Courseware

In principle the next courses should build upon the teaching material provided earlier. The committee would like to receive copies of the material of last year and discuss suitability of the courseware as a starting point for ‘standard’ courseware. (Kees attended Amy’s class, so he is aware of her notes. Piet attended Victor’s class, so he is aware of his notes. Piet has announced the availability of a set of transparencies emerging from the TUE L^AT_EX course.)

Background to courseware are the (emerging) books on T_EX, L^AT_EX and Metafont.

We are not (yet) familiar with the video tapes of TUG.

3.4 Review

In reviewing attention has to be paid to

- **Contents**
Suitability, completeness: exercises and answers. Childs’ selftest schemes should guide a participant to the appropriate level.
- **Pedagogics**
How is it build up? What pedagogical principles are used?
- **Price**
What do the teachers charge for providing there ‘notes,’ in order to circumvent reinventing the wheel.
- **Teachers**
Are teachers willing to work with, or start from, basic material provided by others and refereed by committee?

¹ Background info: Charles R. Martin: T_EX for the T_EXnical Typists. TUGboat 11#3, 425–428.

- **Copyrights.**

To be negotiated by the (NTG) board.

Some teachers (Doug Henderson for example) provide their notes for free, for the best of T_EX etc.

3.5 Tools etc.

Ton modified SliT_EX such that a header containing context sensitive and background information can be supplied. It has not yet been β -tested. Van Oostrum's 'fancy headings' could become a general means for providing context sensitive information as well as background information in the header/footer of transparencies. trspar.sty has not been released for β -testing.

3.6 Course fees

The course fees should be in agreement with those handled by TUG or a private company, and should provide the means to pay:

- the courseware to be supplied,
- the (international) teachers,
- the rent for the (computer) location and infrastructure, the catering,
- the organizational overhead.

The principle to be adhered to is: Those who benefit most (the course takers) should pay, eventually subsidized by NTG for strategic courses.

Because of agreements made with other LUGs and TUG

- Members (of any TUG) enjoy 10% reduction.
- Of every LUG a delegate has to be invited to participate at no costs.

As a rule of thumb: \approx Fl 300,- per day, per participant.

3.7 Organization

For the organization it is best to loosely couple courses to the NTG-days, also in responsibility.

The task to organize courses each year, preferably along with the NTG days, could be delegated to the education committee.

Januari, 1991

Kees van der Laan

Ton Biegstraaten

Piet Tutelaers

4 Appendix

Contribution to TUG BoD

Motion for march TUG BoD meeting:

- adopt the need for an education committee,
- appoint people (to be invited) for the committee,
- thoughts about exams committee(s).

From: Kees van der Laan

Because TeX etc. is the best,
but helaas complex
and full of T_EXfalls
education is paramount.

Another committee: on education

Next to the important committees on the air I like to table the education steering committee.

The task of the committee is

To plan and guard all business related to T_EX etc. education, and report to the board about the matter.

A few questions I have had on my mind rather a long time, might arise awareness of the need for this committee, especially for those not already in favor. I know that education is not much of a respected issue in the USA, and that research has it. I don't agree with that, and . . . in Europe it is different.

The basic idea is that better T_EXnique will be acquired when guided by experienced teachers. It will save time and energy, and generally one will enjoy it, have a nice time and make new friends. I'm self-educated on the issue, it took me a long time, and still have some blind spots, with respect to incompleteness and misconceptions, I guess. Furthermore, THE selfeducation pitfall is too complex T_EXing, opposing the literate programming attitude.

The questions

1. **Why?**
Why are T_EX etc. courses provided? For the proliferation of good T_EXknowledge? For money making?
2. **What price? Discounts?**
At cost? Discount policies are needed.
Teachers and exam committee members have to be budgetted. They are free to donate their salary, though.
3. **For whom?**
Whom is it aimed at? Scientists? Typists? High-school students/teachers? USA-based? Worldwide? I'm not persuading to be exclusive, rather I would favor to plan strategic and attainable goals, in crescendo.
4. **When?**
Just along the main meetings of TUG? Any LUG as well? Demand-driven, though.

5. Where?

Ideally continentwise, but that is also demand-driven.

6. What?

What is the contents of the various modules and how are the modules related to each other, overlap? See prior discussion on that, guarded by Bart Childs. Perhaps Bart could provide the committee with the material he solicited for in Karlsruhe and along with his selftests?

7. Courseware?

We should build upon courseware. Not reinvent the wheel. We should have build upon Samuel's pioneering work. Don't provide another 'Gentle T_EX' and leave it to that, however useful that might be at first sight. We should reiterate, improve, improve, improve, and finally . . . improve. What about the (video) tapes? (See TUGboat ads, e.g. 6#1, p43.) Are they of any use? Experiences? Why did that mechanism not work? Why not sell those tapes at cost? Imagine DeK at home!

8. Examinations?

We should provide for examinations —not just a present-certificate— and make the total *courses+exams* public. LUGs might integrate these courses in their national programs for (non-)regular computer science education. For example as modules in the DTP or EP area. When we have made it available, LUGs can also go ahead and make the best out of it. The above entails appointment of exam committees. Exam committee work is by no means trivial, as I know from experience.

9. Who are the teachers?

We should start with 'teaching the teachers.' I mean provide courses with TUG examinations at the end, in order to qualify for a TUG teacher. (The Dutch bridge union provides similar courses and exams, by no means easy! We also have a national Exam Institute for non-regular computer science education, directly linked to the economics governmental department!) Exams are good, and might stimulate T_EXcognition, and . . . might contribute to respectful T_EXing! We have a bootstrap problem though. Could DeK be of help? Stanford teacher-(summer)-courses?

10. Pedagogics?

Not only T_EXknowledge should be the matter. Attention must be paid to pedagogical principles to be used. TUG courses can be sold under the Gagné pedagogical principles, to name but one that is popular here. It should be part of the teacher-courses.

11. Operational procedures.

Provide operational procedures for the announcements and follow-up of the courses. Evaluation forms. What to do with the answers?

I hope by the above raised questions that everybody will agree on the need for an education committee.

Proposed committee members**Internationally?**

- Pro: worldwide,
 - Con: Long decision lines, too many factors involved.
- Nevertheless, do it that way, but select knowledgeable and efficient working people.

My suggestion for the committee members (I don't know whether those people are willing to accept the invitation to be on the committee): Bart Childs, Doug Henderson, Amy Hendrickson, David Salomon, Malcolm Clark (Cathy Booth? Chris Rowley?), Kees van der Laan, Joachim Schrod?, Stefan von Bechtolsheim?

For the money involved, there should be no problem, I consider investment² in teaching paramount for the future of T_EX.

5 Appendix:**New teacher, T_EXing Math course**

By this note I like to make known to the NTG community that I —Kees van der Laan— am in for teaching. The courses I have on my list are

- T_EX beginners and intermediate, according to Childs' set-up (3 days).
- T_EX topics: T_EXing Math (see set-up below).
- L^AT_EX beginners, demand-driven along the scheme given in de Bruin(1989; Dutch)³ (3 days).

Course set-up: T_EXing Math

The idea is a 3-day course, where the first two days need a classroom with overhead projector. Traditional math typography will be treated from Swanson(1986). Theory will be alternated with exercises, mainly from the T_EXbook. Participants have to work out the exercises in vitro, with paper and pencil. The 3rd day is devoted to hands-on. Participants can get a start with their own publications, if any, otherwise real mathscripts will be provided. PC's (and the projector) are needed for the 3rd day. (I'm familiar with MS-DOS PC's, VAX VMS, UNIX, but . . . I tend to abstract from the hardware.) The material treated is plain oriented. It is basic for T_EXing math via $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX, L^AT_EX, or . . . , whatever!

For whom?

Assumed level of

T_EXknowledge: intermediate, more or less,

²Even better when in phase with other projects like distribution of PD PC T_EXware.

³Bruin, R. de, C.G. van der Laan, J.R. Luyten, H.F. Vogt(1989): Publiceren met L^AT_EX. CWI syllabus 19.

Computer literacy: familiarity with an Operating System, and bring along your favorite editor. It is aimed at authors as well as typists.

Courseware.

The \TeX book.

Swanson, E. (1986): Mathematics into Type. AMS.

Course set-up

- **First day: Morning** (Swanson, TB \approx Chapter 16, 17)
 - Capita from Swanson.
 - Math in text and in display.
 - Math mode, Greek letters, accents, superscripts and subscripts.
 - Context sensitive symbols like openings and closings, varying size.
 - Ordinary formulas: Operands and Operators.
 - Formulas of class 0, empty braces.
 - Expressions (with \wedge , $*$, $/$ (and $\overline{\quad}$), $+$, $-$; functions, openings and closings (a.o. norm fences); integration, summation with and without limits; $\overline{\quad}$ / \atop / \above / \below delimiters; textstyle , scriptstyle , and scriptscriptstyle .)
- **Firts day: Afternoon** (TB \approx Chapter 18)
 - Punctuation.
 - Non-italic letters in formulas. Math function names. $\backslash\text{hbox}$ vs. $\backslash\text{rm}$.
 - Spacing within formulas (automatic, to be suppressed, to be added: table TB170).
 - Dots (ellipses).
 - Line breaking.
 - Lemmas, Theorems and the like.
- **Second day: Morning** (TB \approx Chapter 18, 19)
 - Summary first day (Math atom classes!).
 - Macroscopic aspects:
 - displaylines (free format),
 - alignment without and with numbering,
 - interruption of aligns ,
 - splitting long formulas.
 - Matrices, arguments hypergeometric functions, cases, overbracing/underbracing.
- **Second day: Afternoon** (TB \approx Chapter 19, Appendix B 362, D, E, F)
 - Automatic numbering.
 - Referencing to equations by names.
 - Left justification.
 - Font tables.

Creating new operators ($\backslash\text{buildrel}$, $\backslash\text{mathop}$, and $\backslash\text{mathchoice}$).

Font families of manmac , size-switching macros.

- **Third day:** Hands-on with a real paper.
 - The day will be concluded by mentioning the $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX packages and services, and tailoring your editor into a math \TeX ing intelligent one.

The material to be treated can be adapted on request. The 3rd day could be filled with treatment of plain's math macros, Appendix B-6, p357–362, yielding a complete theoretical course. The other way round, hands-on labs at the first two afternoons as well, is also possible.

Because most, if not all, of the material is in Swanson(1986) and the \TeX book, not much handouts will be needed. AMS folders and packages would be beneficial.

Salary

According to current practice, to be agreed upon.

CV

Studied applied mathematics at the University of Amsterdam. Worked at the CWI for a couple of years, and after that at the computer centre of the University of Groningen. Have been active in writing mathematical software for the NUMAL, IMSL as well as the NAG library, in FORTRAN, ALGOL60/68. Temme and I published a book about evaluating Special Function software. I published articles about mixed-language programming, especially interfacing high-level languages (ADA, PASCAL, ALGOL68, Simula) to FORTRAN (numerical libraries). Have written courseware for FORTRAN, ALGOL68, numerical software, and 'proving-program-correctness' courses.

Since a couple of years active in the EP area. Published with colleagues at the centre a course book on \LaTeX . Have written several reports about \LaTeX , \TeX and SGML. A couple of articles have been published in TUGboat. I was one of the initiators of NTG, and I am its first president. Because of that I was invited to become vice-president of TUG. In 1989, I also developed a one-day SGML course, held at Stanford.

I'm also member of one of the many exam committees of EXIN, the examination institute for non-regular computer science education, resorting under the governmental economics department.

BIJLAGE J

Werkgroep 4: Fonts Activiteiten van de werkgroep

Erik-Jan Vens

Maart 1991

De Font Werkgroep

Fonßs:

Er rijdt in Groningen een bus rond met een duizelingwekkende hoeveelheid bungelende en spartelende vijfen, negens en nullen erop. Mocht iemand mij vragen of er ook negatieve kanten zitten aan het werken met T_EX, L^AT_EX of METAFONT, dan zal ik kunnen antwoorden, dat mijn oog (als aan een spijker) blijft hangen aan dergelijke uitingen van lelijkheid met letters en cijfers.

Wat doet nou zo'n werkgroep

Na een periode van rust is de fontwerkgroep weer ontwaakt.

De werkgroep was heel wat van plan.

De werkgroep is heel wat van plan.

Er een aantal dingen gebeuren op fontgebied, te denken valt aan:

- Het benaderen van ontwerpers om ze ervan te overtuigen dat ze met METAFONT moeten gaan werken. Op deze manier kan de T_EX wereld aan meer 'native'

fonts komen. Het feit dat Hermann Zapf een font voor de T_EXers heeft ontwikkeld, zou als belangrijk punt naar voren gebracht moeten worden. Hierbij staat voorop om te trachten om enkele puur Nederlandse fonts voor T_EX ter beschikking te krijgen.

- Het oprichten van een werkgroep (bijv. via een discussie-lijst of Email) worden, die zèlf een font in METAFONT gaat coderen. Ik voel er zelf veel voor om deze groep op te richten.
- Het trachten te verbeteren van de fonts voor elke type printer. Een voorbeeld van een probleem voor printers met een lage resoluties is dat he vergrotingschema van de Pandora fonts niet goed werkt. Ik ben bezig met het uitwerken van parameterfiles voor deze fonts.
- Het testen van de DC-fonts. DC-fonts zijn de β -test versie van de EC-fonts, de METAFONT files van Norbert Schwartz. In deze fonts zitten alle (nou alle, uh, wel een heleboel) land-specifieke letters. Zoals de n-hacek (ň), die in het Tsjechisch en het Slowaaks voorkomt.

Dit alles staat op het lijstje van de werkgroep fonts. En ook al gaat het niet zo hard als we allemaal wel zouden willen, er komt heus wel wat uit.

Het lettercorps in T_EX

Jos Winnink

Maart 1991

1 Inleiding

In eerste instantie komt de hoeveelheid beschikbare fonts in de *standaard* distributie van T_EX overweldigend over. Een nadere analyse leert dat door de manier waarop met deze informatie wordt omgesprongen een aanzienlijke hoeveelheid informatie redundant wordt opgeslagen. Tevens moet worden opgemerkt dat het lineair schalen van fonts op zijn hoogst als een laatste redmiddel moet worden gebruikt en niet voor het creëren van ontbrekende corpsen.

Mede op basis van opmerkingen in de literatuur is een analyse gemaakt wat standaard beschikbaar is in T_EX en wat gewenst zou zijn. In het vervolg van dit artikel zal worden aangetoond dat er nogal wat te verbeteren valt zonder de vrijheid van de gebruiker met betrekking tot het gebruik van fonts aan te tasten.

2 Het lineair schalen van fonts

T_EX biedt standaard mogelijkheden om lettertypes in verschillende corpsen te maken door uitgaande van de ontwerp-grootte het *font* lineair te vergroten door middel van de commando's:

- at
- \magnification
- \magstep
- scaled

Het vergroten van fonts wordt in T_EX op twee manieren gebruikt en wel om:

- een document in zijn geheel te schalen, dit gebeurt met behulp van het T_EX commando `\magnification`, dat overigens in L^AT_EX buiten werking is gesteld.
- fonts te creëren in een corps dat niet aanwezig is.

2.1 Globaal schalen

Het globaal vergroten van een document wordt gebruikt om het oplossend vermogen van een output device maxi-

maal te kunnen benutten door eerst vergroot *af te drukken* en later (fotografisch) te verkleinen.

De vraag doet zich voor of dit gebruik van vergroten niet door de DVI-driver zou moeten gebeuren het heeft in feite niets met het document te maken. Veel DVI-drivers ondersteunen het vergroten van een document, waarvoor overigens wel lineair geschaalde fonts ter beschikking moeten staan. In feite zou het globaal schalen in DVI-drivers kunnen worden ingebouwd door een algoritme analoog aan het algoritme dat METAFONT gebruikt te gebruiken.

2.2 Creëren van ontbrekende corpsen

Met behulp van de genoemde commando's en een goed gekozen verzameling vergrotingen lijkt het in de praktijk niet/nauwelijks voor te komen dat een lettertype in het gewenste lettercorps niet aanwezig is. Een vaak gehanteerde methode is dan ook om alle basiscorpsen in elke denkbare vergroting ter beschikking te stellen.

Of de aldus te verkrijgen fonts allemaal typografisch verantwoord zijn staat nog te bezien. In zijn algemeenheid moet voor elk corps een lettertype opnieuw worden ontworpen. Met name bij wat extremere vergrotingen ontstaan onevenwichtige letterbeelden door de lineaire schaling; de letters worden te breed, de zwarting veranderd met als gevolg dat de leesbaarheid verminderd.

In het vervolg wordt de in T_EX gehanteerde methode geanalyseerd en wordt een aanzet gegeven voor een andere (betere) opzet.

3 De situatie in T_EX

Wat is nu de situatie in *standaard* T_EX? Uitgaande van het CMR-lettertype in de corpsen 5, 6, 7, 8, 9, 10, 12, 17 punten¹ levert dit voor de 7 vergrotingsmaten s0, sh, s1, s2, s3, s4, s5 (: \magstep 0 tot en met \magstep 5) de volgende beschikbare corpsen op:

¹ Alhoewel er een verschil is tussen DIDOT-punten en PICA-punten is dat voor dit verhaal niet relevant en wordt in dit verhaal impliciet uitgegaan van PICA punten. Voor een systeem dat gebaseerd is op een ander maatsysteem kan een analoge redenering worden gehouden.

lettertype	\magstep						
	s0	sh	s1	s2	s3	s4	s5
cmr5	5.0	5.5	6.0	7.2	8.6	10.4	12.4
cmr6	6.0	6.6	7.2	8.6	10.4	12.4	15.0
cmr7	7.0	7.7	8.4	10.1	12.1	14.5	17.4
cmr8	8.0	8.8	9.6	11.5	13.8	16.6	19.9
cmr9	9.0	9.9	10.8	13.0	15.6	18.6	22.4
cmr10	10.0	11.0	12.0	14.4	17.3	20.7	24.9
cmr12	12.0	13.1	14.4	17.3	20.7	24.9	29.9
cmr17	17.0	18.6	20.4	24.5	29.4	32.3	43.3

Uit deze tabel wordt duidelijk dat er een groot aantal (bijna) dubbelstellingen ontstaan. Oorzaak is de keuze van de basiscorpsen gekoppeld aan het lineaire schalingsmechanisme. De esthetische kanten van het lineair vergroten worden voor het gemak maar even buiten beschouwing gelaten.

4 Welke lettercorpsen zijn gewenst?

Huib van Krimpen² merkt op (blz 310) dat een overzicht van de verschillende corpsen er als volgt uit zou kunnen zien:

5 = 5@s0	5½ = 5@sh	6 = 6@s0	6½ = 6@sh
7 = 7@s0	7½ = 7@sh	8 = 8@s0	8½ = 7@s1
9 = 9@s0	9½ = 8@s1	10 = 10@s0	10½ = 6@s3
11 = 11@s0	11½ = 8@s2	12 = 12@s0	
13 = 9@s2	14 = 8@s3	15 = 6@s5	16 = —
17 = 17@s0	18 = —	19 = 9@s4	20 = 8@s5
21 = 10@s4	22 = 9@s5	23 = —	24 = 17@s2
26 = —	28 = —	30 = 12@s5	32 = 17@s4
34 = —	36 = —	38 = —	40 = —
42 = 17@s5	44 = —	46 = —	48 = —
50 = —	52 = —	54 = —	56 = —
58 = —	60 = —		

Het blijkt uit deze gegevens dat van de 56 fontfiles (PK-files), die \TeX ter beschikking staan er slechts 27 gebruikt kunnen worden om lettercorpsen die voldoen aan de reeks van van Krimpen te genereren.

Overigens moet hierbij worden opgemerkt dat een deel van deze 27 corpsen wordt gegenereerd door middel van een aanzienlijke vergroting, hetgeen de letterbeelden zeker niet ten goede komt. Documenten met combinaties van corpsen zullen een zeer onevenwichtige indruk maken. Als voorbeeld mag dienen de combinatie 10 punts en 15 punts corps. Hierbij is het 10 punts corps een basisgrootte (aanwezig in $\backslash\text{magstep}0$), het 15 punts corps dient echter geconstrueerd te worden uit het 6 punts basiscorps vergroot tot $\backslash\text{magstep}5$, die resulteert in een combinatie met een goed ontworpen font (10 punts) en een ernstig *misvormd* font van corps 15.

In de vorige paragraaf is aangetoond dat de helft van de

- 6 tot en met 12 punten, in stappen van een $\frac{1}{2}$ punt.
- 13 tot en met 24 punten, in stappen van 1 punt.
- 26 tot en met 60 punten, in stappen van 2 punten.

Welke zinvolle lettercorpsen kunnen we nu uit de tabel in de vorige paragraaf afleiden die overeenkomen met de opmerking van van Krimpen? We voegen de corpsen 5 en $5\frac{1}{2}$ toe, omdat deze in \TeX op diverse plaatsen gebruikt worden. In onderstaande tabel staat aangegeven hoe in \TeX met behulp van het standaard mechanisme de gewenste corpsen al dan niet te maken zijn.

fontfiles zinloos wordt gegenereerd en opgeslagen. Om over alle corpsen uit de *van Krimpen*-reeks te beschikken zijn 45 PK-files nodig. Beperken we ons tot corps 42 dan kan worden volstaan met 36 PK-files.

5 Idee

Het idee is om de corpsen te genereren volgens de reeks van van Krimpen in de vergroting $\backslash\text{magstep}0$.

De voordelen zijn duidelijk. Er wordt een typografisch meer verantwoord produkt gemaakt met behulp van \TeX , waarbij een schijnbare flexibiliteit, die leidt tot overbodige opslag van gegevens, wordt vermeden.

Welke problemen brengt het invoeren van de *van Krimpen* reeks ten opzichte van het standaard \TeX -mechanisme met zich mee?

²H. van Krimpen, *BOEK over het maken van boeken*, Gaade uitgevers c.v., Veenendaal, 1986²

- Een groot aantal fonts zal opnieuw moeten worden ontworpen. Dit komt er op neer dat de parameters in de MF-driver file's moeten worden bekeken om na te gaan welke aanpassingen nodig zijn. Overigens met behulp van de overzichten in VOLUME-E (CM-fonts), moet dit niet al te veel werk zijn. Ook de parameterisering van John Sauter kan met name voor de CM-fonts handig zijn.
- Er zijn (veel) meer TFM files nodig in het gebruik en wel één voor elk lettertype in elk corps.
- Het is nodig om ervoor te zorgen dat standaard \TeX bestanden, die gebruik maken van de vergrotingsmogelijkheden, kunnen blijven werken door alles om te rekenen naar corpgroottes uit de genoemde reeks. Zonodig moet de corpgrootte bepaald worden door afronding. In combinatie met het fontselectiemechanisme van Mittelbach/Schöpf kan dit voorstel zonder meer gerealiseerd worden. Vermoedelijk moet voor de onesthetici onder de \TeX -gebruikers een mogelijkheid worden ingevoerd om vergroote misvormde fonts te kunnen gebruiken. Bij gebruik van *plain* \TeX en \LaTeX moeten er wijzigingen worden aangebracht in de basissystemen. Ook moet het `\font` commando van \TeX worden vervangen door een gewijzigde versie.

- Een afzonderlijk op te lossen probleem vormt het globale vergroten van een document met behulp van `\magnification`. Voor deze toepassing zou de huidige methode gehandhaafd kunnen blijven, waarbij wel moet worden nagedacht over de te gebruiken vergrotingen.

6 Conclusie

De mogelijkheden die \TeX biedt om fonts te schalen lijkt bruikbaar dan ze in feite is en maakt de situatie voor de (beginnende) gebruikers vaak ondoorzichtig. De ondoorzichtigheid komt mede voort uit het feit dat bijvoorbeeld \LaTeX geschaalde fonts gebruikt. Door de manier waarop de fonts worden gebruikt wordt veel redundante informatie opgeslagen en de door vergroting verkregen lettertypes missen het raffinement van de goed ontworpen basisletters.

Als het voorstel om te komen tot een standaardreeks van corpgroottes wordt uitgevoerd en daarvoor de benodigde ondersteuning in \TeX wordt ondergebracht is het mogelijk om typografisch meer verantwoorde resultaten te verkrijgen, terwijl de benodigde diskruimte vermoedelijk kleiner zal zijn.

BIJLAGE K

Werkgroep 6: Lijst en link met fotozetters

T_EX in hoge kwaliteit

Theo A. Jurriens

Maart 1991

Op verzoek van de NTG najaarsvergadering heb ik eind november een vraag het net opgestuurd met als doel het inventariseren van beschikbare hoge kwaliteit PostScript printers. Zoals ik verwachtte was de respons niet erg hoog. Auteur is niet verantwoordelijk voor genoemde bedragen, **maak afspraken over de condities**.

Een overzicht:

Drukkerij Plantijn

De drukkerij beschikt over een fotozetmachine die PostScript kan verwerken. De test-invoer was een PostScript file van T_EX, waarbij van de PostScript fonts gebruik werd gemaakt, en waarin ook een aantal T_EX "rules" zaten van een tabel. De test was geslaagd, kosten ongeveer fl. 25,- per pagina op fotopapier (is afhankelijk van de grootte van de opdracht) aldus Dick van Soest (Universiteit Twente).

Schrijen-Lippertz

Deze zetters/drukkerij in het zuiden van het land heeft lange traditie inzake T_EX. Aanvoer via modem is mogelijk. Aanleveren op zowel 5.25 en 3.5 inch diskettes is geen probleem. De prijs bedraagt ± fl. 12.50 per pagina.

Transcripta

Het bedrijf beschikt over een Linotronic 100. A-4 prijs bedraagt fl. 8,- exclusief verzendkosten. Aanleveren op zowel 5.25 en 3.5 inch diskettes is geen probleem. Aanvoer via modem is mogelijk. In verband met eigen fonts of logo's is meeleveren .MF files geen probleem.

Universiteitsdrukkerij, Groningen

De drukkerij beschikt over een Varityper 600 (600 dpi). Pagina prijs fl. 1.50, aanlevering bijvoorkeur op 5.25 inch, 360 kbyte diskette. Aanlevering (.TEX file, .PS

file) via het net is mogelijk taj@rug.nl, eigen logo's, fonts is geen probleem.

Adressen:

Drukkerij Plantijn BV

M.C. v.d. Velden
Molenlaan 19
2908 LL Capelle a.d. IJssel

Drukkerij Schrijen-Lippertz

Frans de Bruijn
Postbus 23002
5367 ZG Voerendaal
tel. +31 45 751630

Transcripta

J. Scholten
Hemweg 6
1608 HL Wijdenes
tel. +31 2293 1795

Universiteitsdrukkerij

De heer Aldershof
Blauwborgje 2c
9747 AC Groningen
tel. +31 50 635532/5530

Dick van Soest

University of Twente
Computer Science Department
P.O. Box 217
7500 AE Enschede
Tel. +31 53 893736/893690
Internet: vansoest@cs.utwente.nl

BIJLAGE L

Werkgroep 7: PC-zaken

T_EX voor MS/PC-DOS PC's

Jos Winnink

Maart 1991

Sinds het vorige verslag van de werkgroep 7 (PC zaken) zijn er geen schokkende gebeurtenissen voorgevallen. Inmiddels is vergelijkingsmateriaal met betrekking tot de snelheid van de verschillende T_EX implementaties voor de MSDOS-PC's beschikbaar. Het betreft de artikelen van Erich Neuwirth in TUGboat, Volume 11 (1990), Nr. 4, p. 578 ev. en in *Die T_EXnische Komödie* van december 1990, p. 37 ev. van de hand van Joachim Lammarsch.

Op basis van de gegevens uit het vorige verslag van de werkgroep en de gegevens uit bovengenoemde artikelen kan het volgende worden opgemerkt. Er zijn nu gegevens beschikbaar voor de commerciële pakketten PCT_EX, TurboT_EX en μT_EX en voor de *Public Domain* DOST_EX, emT_EX, PubliCT_EX, sbT_EX. Het combineren van de gegevens uit deze drie overzichten moet met de grootste omzichtigheid gebeuren omdat de gebruikte testbestanden niet dezelfde zijn.

Als alle gegevens worden gecombineerd komt het volgende beeld naar voren:

1. de systemen met de beste performance zijn: emT_EX, μT_EX en sbT_EX. De onderlinge positionering van deze drie produkten is wat afhankelijk van de gebruikte invoer. Het meest constant in prestaties lijkt emT_EX te zijn. De onderlinge verschillen tussen deze implementaties zijn niet erg groot, maar over het algemeen lijkt emT_EX de snelste versie.
2. PCT_EX lijkt over het algemeen wat trager te zijn dan de drie produkten uit groep 1.
3. DOST_EX en TurboT_EX zijn doorgaans een factor vier of meer langzamer dan de overige versies. Ook zijn deze implementaties erg gevoelig voor de beschikbare hoeveelheid vrij geheugen.

Mijn conclusie uit het vorige verslag wordt bevestigd door de andere twee bronnen. emT_EX is een aantrekkelijk produkt mede door het feit dat het gedistribueerd wordt in combinatie met printer driver's, screen previewer, T_EXCAD, METAFONTe.d..

De waarde van emT_EX wordt ook door anderen onderkend, die dan ook specifieke emT_EX uitbreidingen ondersteunen. Met name de POSTSCRIPT driver (DVIPS) van Tom Rockicki is hier een voorbeeld van. DVIPS ondersteunt de nieuwe versie van de *fontlibraries* (.fli files) alsmede een aantal *specials* van emT_EX.

Omdat ondergetekende zelf een goede POSTSCRIPT driver nodig had en een '.exe' file voor MS-DOS van de DVIPS driver op dat moment niet voorhanden was is er op basis van de DVIPS-distributie een dergelijke file gemaakt. Om overbodig werk te voorkomen is een '.exe' van DVIPS 5.47 en bijbehorende programmatuur naar Piet van Oostrum gestuurd, alwaar deze in de archieven is opgenomen.

De *schoenendoos*¹ met emT_EX is eind november 1990 aan zijn reis begonnen langs de belangstellenden en heeft deze inmiddels (nagenoeg) voltooid. Op basis van emT_EX zal een *N T G* distributie van T_EX voor MS-DOS gemaakt worden.

Jos Winnink,

privé:

Valkhof 54,
2261 HS Leidschendam
070 - 3 20 43 81

werk:

Centraal Planbureau,
van Stolkweg 14,
2585 JR Den Haag
070 - 3 51 41 51
winnink@ecn.nl

¹ volgens Nico Poppelier

Verkrijgbaarheid emTeX²

Pieter Bison

Maart 1991

Er zijn een aantal mogelijkheden om de emTeX diskettes te (laten) kopiëren:

- Via de NTG. Neem contact op met Jos Winnink: winnink@ecn.nl
- Via Aston University.³
- Ikzelf kan kopieën op 5 1/4 HD diskettes maken. Hieronder heb ik m'n aankondiging van oktober vorig jaar herhaald.

EmTeX is nu beschikbaar in een nieuwe distributie (gedateerd 25 september 1990). De belangrijkste verschillen met de vorige versie zijn:

- METAFONT 2.0 en TeX 3.0: 8-bits invoer, afbreken voor meer talen tegelijk etc..
- Virtual (composite) fonts worden ondersteund.
- Driver voor HP Deskjet.
- SliTeX fonts.
- Nieuw fli-format voor font-bibliotheken.
- Meeste documentatie nu ook in het Engels.

EmTeX is een volledige TeX-implementatie voor MS-DOS/OS-2 PC's en bevat naast TeX en METAFONT (ook in 'grote' versies) o.a.: MakeIndx, BibTeX, PiCTeX, printerdrivers, Screen-previewer, alle TeXware- en MFware- programma's en fonts voor vele printers. Daarnaast alles voor L^ATeX en de Computer Modern METAFONT sources. Er is een aantal nuttige utilities bijgeleverd. Om er twee te noemen:

- **MFjob:**
'geautomatiseerd' fonts produceren.
- **TeXcad:**
interactief plaatjes maken voor L^ATeX picture omgeving.

U denkt wellicht: Waarom omschakelen? emTeX doet nu toch alles wat ik wil. Schakel a.u.b. toch over op de nieuwe versie omdat anders één van de grote voordelen van TeX, de universele compatibiliteit, verloren dreigt te gaan. In de woorden van Knuth:

'Let us root out and destroy the obsolete 7-bit systems, even though we were able to do many fine things with them.'

De volgende pakketten zijn beschikbaar:

emTeX (compleet) 6 diskettes 1.2Mb

Fonts:

HP-LaserJet+	(300 dpi)	5 diskettes 1.2Mb
EPSON FX-80	(240x216 dpi)	4 diskettes 1.2Mb
NEC P6 low	(180 dpi)	3 diskettes 1.2Mb
NEC P6 medium	(360x180 dpi)	4 diskettes 1.2Mb
NEC P6 high	(360x360 dpi)	6 diskettes 1.2Mb
Apple Imagewriter	(160x144 dpi)	3 diskettes 1.2Mb

Naast deze printers worden de volgende apparaten ondersteund (gebruiken een van bovenstaande pakketten):

300 dpi	:	HP-Deskjet, Kyocera F-1010,
240x216 dpi	:	Tandy DMP-130, IBM Proprinter 4201/2
180 dpi	:	Epson LQ serie, Panasonic KX P1124, IBM Proprinter 4207/8
360x180 dpi	:	Epson LQ serie, Panasonic KX P1124, IBM Proprinter 4207/8
360 dpi	:	Epson LQ-550, Panasonic KX P1124
160x144 dpi	:	C.Itoh 8510A

Verder worden de brede versies van Epson LQ en NEC (P7) ondersteund. Er is een dvi naar msp/pcx omzetter en een previewer voor Viking I beeldschermen. Voor de gewone previewer is een van bovenstaande hoog-resolutie (300 of 360 dpi) sets nodig. De previewer ondersteunt bijna alle videostandaarden.

De font-pakketten bevatten de volledige Computer Modern familie in 7 vergrotingsstappen. De fonts zijn opnieuw gegenereerd maar vrijwel gelijk aan de vorige distributie; alleen de SliTeX fonts (voor sheets) zijn toegevoegd.

Wat niet overeenstemt is het formaat van de font-bibliotheken (fli-files) die door de emdrivers gebruikt kunnen worden. Oude fli-files kunnen niet gebruikt worden door nieuwe drivers en nieuwe fli-files niet door oude drivers. Wel kunnen oude fli-files omgezet worden naar het nieuwe formaat. Bij aanvraag van alleen de nieuwe emTeX kan dit gebeuren m.b.v. een bijgeleverde batch-file. (fli's van nieuw naar oud omzetten is ook mogelijk maar niet aan te raden: schakel dan meteen over op de nieuwe emTeX).

emTeX is op een aantal ftp-adressen op te halen:
- rusmv1.rus.uni-stuttgart.de [129.69.1.12]

²Deze informatie werd op 19 maart 1991 via TeX-NL verspreid

³Zie in de hierna komende bijlage van Kees van der Laan 'Enige suggesties voor WG-PC's'

- (in: soft/tex/emtex)
- terminator.cc.umich.edu [35.1.33.8]
(in: msdos/text-mgmt/TeX/emtex)
 - eba.eb.ele.tue.nl [131.155.2.25]
(in: tex/emtex.new)
 - archive.cs.ruu.nl [131.211.80.5]
(in: TEX/emtex)
- (grote hoeveelheden bij deze laatste alleen tussen 18.00 en 8.00)
- en daarnaast ook op de Aston server.

Voor degenen die hiervan echt geen gebruik kunnen maken is er de mogelijkheid om diskettes te laten kopiëren.

Stuur:

- Het vereiste aantal geformatteerde en gelabelde HD 1.2 Mb, 5 1/4 diskettes. Andere diskette-formaten zijn uitgesloten. Vul de labels a.u.b. zelf al in (bijv. emTeX 1/6 etc.)
- Een volledig geadresseerde (d.w.z. met adres geadresseerde (uzelf) en dat van afzender (mijn adres)) retour-

envelope met voldoende porto. Gebruik voor de porto alleen normale zegels, geen stempels (zijn gedateerd).

aan:

Pieter Bison
C. Springerstraat 11-2
1073 LD Amsterdam.

De pakketten worden alleen als een geheel verzonden. Vragen over het functioneren van emTeX graag naar een van de TeX lists (UKTeX, TeXhax of TeX-NL) of in laatste instantie naar de maker van dit alles:

Eberhard Mattes
Teckstrasse 81
7141 Moegligen
Duitsland
mattes@azu.informatik.uni-stuttgart.de

(Voeg ter beantwoording een aan uzelf geadresseerde retourenvelope toe met porto in de vorm van internationale antwoordcoupons)

BIJLAGE M**Enige suggesties voor WG-PC's****Kees van der Laan**

Maart 1991

Eén van de belangrijkste zaken om het gebruik van T_EX etc. te bevorderen is een goede PD PC versie, met goede documentatie, te hebben. Niet allen beschikbaarheid is van belang, maar het moet ingeblikt, in handige setjes, op de schappen liggen. Ook dienen de bestellingsprocedures professioneel afgehandeld te kunnen worden. Maar ja wij zijn vrijwilligers, nietwaar? Om toch! Niets is frustrerender en afstotender dan bedolven te worden onder een berg floppies die niet werken, nadat men geprobeerd heeft het aan de praat te krijgen door te zoo-en, te selecteren, hergroeperen, schrappen van het overbodige, en tenslotte tegen de beperkingen van de onderhavige PC te lopen.

Kortom, subsetting met duidelijke documentatie en werkende voorbeelden van installatie en gebruik, afgerond met professionele logistiek, en ondersteund door het cursusgebeuren!

Bovendien moeten wij niet vergeten dat de markt die wij als eerste proberen te 'veroveren' —de Nederlandse wiskundigen— niet geïnteresseerd is in de in-en-outs van PC's noch van T_EX. Dat zijn slechts beperkte hulpmiddelen, waar het publiceren helaas mee moeten doen.

De suggesties

- naar de werkgroep toe: maak kant-en-klare subsets etc, en
- naar de NTG toe: organiseer professionele verspreiding.

Het tij is gunstig ervoor, daar de Board of Directors (BoD) van TUG in Maart besloten heeft ook PD T_EX distributie te verzorgen. DANTE, GUTenberg en ukT_EXug doen het al.

Ter illustratie het volgende uit ukT_EX.

emTeX (for MS-DOS): Send 11 UNFORMATTED 1.44Mb or 18 720K 3.5" disks, or 12 UNFORMATTED 5.25" disks, with return postage.
emTeX is now distributed with a copy of 'Baskerville' and documentation. Please therefore send the disks in packing of A4 size or enclose an A4 envelope.

*** Postage rates: all prices in pounds sterling ***
(make cheques payable to Aston University)

0.5" tapes: UK: 2.50 (one tape), 5.00 (two tapes).
Europe: 5.00 (one tape), 9.00 (two tapes).
Outside Europe please enquire.

8mm tapes: UK: 1.00, Europe: 2.00.

DC600A cartridges:

UK: 1.00, Europe: 2.00.

Diskettes:

Quantity/Size	Europe	World	UK 1st	UK 2nd
18/3.5"	3.10	5.10	1.40	1.10
11/3.5"	1.80	2.90	0.80	0.65
18/5.25"	1.20	2.00	0.60	0.50
11/5.25"	0.80	1.30	0.50	0.35

*** Postal address ***

Peter Abbott,
Computing Service, Aston University, Aston Triangle, Birmingham B4 7ET

(Please include SELF-ADDRESSED ADHESIVE LABELS for return postage).

Figure 1: Distribution Ad from ukT_EX.

Uit het ingeslotene blijkt dat de logistiek goed verzorgd is. Het probleem van virussen hebben zij hopelijk ook onder controle.

Wat ik mis alom is de subsetting idee, en de aandacht voor de echt kleine $\text{T}_{\text{E}}\text{X}$ gebruiker, zoals ik tegenwoordig. Wederom een citaat (TB350):

‘First, it is desirable to keep the total number of fonts of plain $\text{T}_{\text{E}}\text{X}$ relatively small, because plain $\text{T}_{\text{E}}\text{X}$ is a sort of standard format; it should not cost much for someone to acquire all the fonts of plain $\text{T}_{\text{E}}\text{X}$ in addition to those he really wants, . . .’

Indachtig de ‘core+modules’-idee, het volgende voorstel, om mee te beginnen:

Basis: plain- $\text{T}_{\text{E}}\text{X}$, version π ,

(Niets meer en niets minder. Welke? EM- $\text{T}_{\text{E}}\text{X}$!)

$\text{T}_{\text{E}}\text{X}$ uitbreidingen (ieder een zelfstandig geheel):

1. manmac.tex (DeK’s $\text{T}_{\text{E}}\text{X}$ book macros en fonts).
2. Proefschriften formats voor het Engels, met wat sjablonen.
(Een aantal gangbare. Welke-o-welke?)¹
3. Tijdschriften formats.
(In ieder geval TUGboat, en die van AMS, . . .)
4. AMS package voor $\text{T}_{\text{E}}\text{X}$, fonts en formats.
5. (EMACS) $\text{T}_{\text{E}}\text{X}$ intelligente editor, met wat gebruikersinterfaces vooral geënt op bovenstaande formats.
6. $\text{P}_{\text{I}}\text{C}_{\text{T}}\text{E}_{\text{X}}$ of iets dergelijks.
(Niet PD? In eerste instantie weglaten? Lijkt mij niet.)
7. μ -spell, spellingschecker.
(Niet PD, maar toch maar proberen.)

$\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}3.?$

1. $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ zelfe
2. Proefschrift stylen voor het Engels.
(Een aantal gangbare plus sjablonen.)¹

3. Nederlandse stijlen: brief, rapport, artikel,
4. Tijdschrift stijlen.
(In ieder geval TUGboat, en die van AMS)
5. AMS package voor $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, fonts en formats.
6. (EMACS) $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ intelligente editor, met wat gebruikersinterfaces.

Verder zou het mooi zijn om disjunct daarvan font collecties te hebben (Adobe’s? Postscript? en enige speciale zoals schaafonts, dingbats) Meer iets van later orde?

Het zou mooi zijn als bovenstaande als beleid wordt nagestreefd en dat gesouffleerd door de prioriteiten handige setjes ‘uit de schappen’ doorgestuurd zouden kunnen worden, zowel op 1.4 als .36MByte floppies. Soortgelijk voor andere systemen zoals Atari en Macintosh. Een identieke opzet op de server zou de eenvoud dienen.

Vanuit energetisch standpunt is het van belang te synchroniseren en in synergie te resulteren.

Stel je voor dat in de besprekingen met het CWI en het Wiskundig Genootschap wij kunnen zeggen:

Kijk, naast de riant beschikbaarheid van $\text{T}_{\text{E}}\text{X}$ etc. op de werkplek, kan men het ook thuis gebruiken. Wij hebben namelijk diverse sets voor PC’s, gratis, nou ja op basis van verzendkosten. Opleidingen? Geen probleem. $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, $\text{T}_{\text{E}}\text{X}$ etc. Speciaal voor wetenschappelijke typistes? U zegt het maar en wij draaien, zullen wij de agenda even trekken, wanneer? Of wilt U liever een videotape-je? Met de meester als docent?

Kortom als de PC- en educatie-activiteiten elkaar aanvullen c.q. ondersteunen, en als het wervingsbeleid naadloos aansluit, dan zijn wij toch aardig bezig, zou ik zeggen.

En dat met een handje vol.

¹ Het is voorlopig niet nodig Nederlandse varianten te hebben, omdat de proefschriften, m/n/ de β -wetenschappelijk, in het Engels zijn.

BIJLAGE N**Werkgroep 8: T_EX Conferentie Nederland****Joop van Gent**

Maart 1991

Op de vorige vergadering is aan Joop van Gent gevraagd hoe het zit met de kansen om op de universiteit in Tilburg (KUB) een NTG-conferentie te organiseren. Hij heeft toen als antwoord gegeven dat hij een dergelijk congres zou willen organiseren onder twee voorwaarden:

1. het congres zou een bredere thematiek moeten hebben dan alleen T_EX, zelfs breder dan typesetting, namelijk *geïntegreerde document systemen* (hieronder nader te specificeren);
2. een onderzoeksvoorstel van zijn hand, met als onderwerp document retrieval en natuurlijke taal, zou door SPIN/MMC gefatteeerd moeten zijn.

Aan de tweede voorwaarde is inmiddels voldaan. Het idee van een congres is aangezwengeld en door zowel SPIN/MMC medewerkers als het Instituut voor Taal- en Kennistechnologie (ITK) goed ontvangen. Hoewel de details uiteraard nog nader uitgewerkt moeten worden, staat reeds het volgende op papier:

1. **Congres:**
DOCUMENT SYSTEMEN
2. **Datum:**
Ergens in 1992
3. **Doel:**
Het congres is bedoeld om een aantal prominente onderzoekers/ontwikkelaars op het gebied van document systemen bij elkaar te brengen. Het is ook de bedoeling het bedrijfsleven te betrekken, in ieder geval zo mogelijk voor sponsoring, wellicht ook in de vorm van sprekers.
4. **Opzet:**
De idee op dit moment is om het congres één dag te laten duren en 'invited speakers' een praatje te laten houden. Geen paper submission dus (te veel georganiseer vindt men hier), maar daarover valt natuurlijk

te praten.

5. **Onderwerpen:**

- a. document productie-systemen
 - spellings- en stijlcorrectoren
 - afbreekalgoritmen (H&J)
 - thesaurussystemen
 - auteurssystemen
 - typesetting
- b. information retrieval waarbij de informatie in documenten is opgeslagen
 - indexering m.b.v. linguïstische kennis
 - bibliografieën
 - full text retrieval m.b.v. linguïstische kennis
 - logische structuren in documenten
 - image handling
- c. interfaces naar databases met documenten
 - query languages, windowing, menu's
 - multimodale interfaces
 - natural language interfaces
- d. document management
 - het beheersbaar maken van informatie in vluchtige documenten
 - koppeling van documenten aan anderssoortige databases: life links
 - beveiliging

Nu wil het toeval dat ondergetekende per 1 mei a.s. van baan verandert (TNO), hetgeen betekent dat mijn rol in de organisatie vanuit de universiteit in ieder geval zo goed als verdwijnt. Binnen het ITK zijn echter twee andere mensen gevonden die naast Huub Mulders en eventuele andere NTG-leden graag actief willen zijn in de organisatie. Hun namen: Hans Pajmans (paai@kub.nl) en Arthur van Horck (horck@kub.nl).

BIJLAGE O**WG10: Imposing structure upon \TeX**

—SGML flavored—

Kees van der Laan

March 1991

Introduction

After my SGML- \TeX presentation at the fall 90 NTG meeting some e-mail discussion arose.

Ton's reaction to Sake's afterthoughts contains a.o. the statement

' ... \LaTeX (and \TeX) lack the semantic properties to enforce compliance to a structure ... '

Well, that is not true. Simple counterexamples are: \LaTeX 's title environment, which is rather restricted and another is the shielding of \TeX commands from \LaTeX , but those are elementary redeclarations.

A conclusion of my work was that this area needs more development/research, especially how to add to a format/style in a simple and robust way, compliance enforcing code. From the discussion I understand that that point has been missed completely. I don't know yet whether \TeX is capable of providing full generality, whatever that may be.

Imposing structure

In order to make it more clear let us concentrate on some simple examples. Suppose a DTD consists of

- **an element, S(quence),**
which consists of two elements which have to be used in the given sequence order
<!ELEMENT S -- (a, b)>
- **an element, O(r),**
containing a and b, which can be used in any order
<!ELEMENT O -- (a|b)>

Furthermore, a and b may contain character data
<!ELEMENT (a, b) -- CDATA>.

The above means that correctly marked up copy complies with.

SGML marked up	\TeX marked up
<S><a>a text	\bS\bba a text\ea
b text	\bb b text\eb
</S>	\eS
and	
<O><a>a text	\bO\bba a text\ea

```
</O>                                \eO
as well
<O><b>b text</b>                       \bO\bba b text\eb
</O>                                \eO
```

Below the Sequence and Or structures are separately elaborated.

Sequence

The idea is to have counters and whenever a or b are entered to increment the ca, respectively cb counter. Furthermore when b is entered `aprecb := (ca=1)`. At the end of S `ca = 1 \wedge cb = 1 \wedge aprecb` is checked. It is not elegant to have b spoiled with code in order to verify whether a preceded, especially when b is to be used in other structures as well.¹

The sequence structure can be imposed as shown by the following \TeX ing.

```
%!ELEMENT S -- (a,b)>
%!ELEMENT (a|b) -- CDATA>
%implemented in TeX via
\newif\ifSenv \Senvfalse %S environment
\newif\ifaprecb\aprecbfalse%a precedes b?
\newcount\ca
\newcount\cb
\def\bS{\bgroup\ca0\cb0
        \Senvtrue}%end bS
\def\eS{%Check
%ca=1 and cb =1 and \aprecbtrue,
\ifnum\ca=1
  \ifnum\cb=1
    \ifaprecb %Ok
      \else\errmessage{At end
        Seq, a did not precede b}
    \fi
  \else\errmessage{At end Seq,
        ca=1 and cb<>1}
  \fi
\else
  \errmessage{At end Seq ca<>1}
\fi\egroup}%end eS
\def\ba{\advance\ca1}
\def\ea{}
\def\bb{\ifnum\ca=1 \aprecbtrue\fi
        \advance\cb1}
\def\eb{}
```

¹ Andrew suggested either to use the `\let` mechanism, or even better to take advantage of \TeX 's stacking. See the appendix.

Or

Again counters are used. At the end of Or ($ca = 1 \wedge cb = 0$) \vee ($ca = 0 \wedge cb = 1$) is checked. The \TeX ing reads.

```
%<!ELEMENT O -- (a|b)>
%<!ELEMENT (a|b) -- CDATA>
%implemented in TeX via
\newcount\ca
\newcount\cb
\def\b0{\bgroup\ca0\cb0}%local cond
\def\e0{%check:
%(ca=1 and cb=0) xor (ca=0 and cb=1)
\ifnum\ca=1
  \ifnum\cb=0 %Ok
  \else\errmessage{Or element
    ended with ca=1 and cb<>0}
  \fi
\else
  \ifnum\ca=0
  \ifnum\cb=1 %Ok
  \else\errmessage{Or element
    ended with ca=0 and cb<>1}
  \fi
\else\errmessage{Or element
  ended with ca<>0, 1 }
\fi
\fi
\egroup}%end e0
%
\def\ba{\advance\ca1}
\def\ea{}
\def\bb{\advance\cb1}
\def\eb{}
```

Combinations

Generalization and combination of the above could yield a document structure which might have repeated Or-paths or Sequence-paths: for example Or-Or-Seq-Or-...

Three possibilities (a|b|c) can be composed from ((a|b)|c) and similarly three in a row (a, b, c) from ((a, b), c), both are repetitions. The optional occurrence of an element, say d, is just an Or of d with empty, (|d). Check for repetitions ask for appropriate counter checking.

I believe that a format can be build from these elements conforming to an SGML DTD, because a DTD prescribes essentially a tree, with repetitions. Insertions and the like are not considered for the moment.

I don't consider this difficult to elaborate in principle, but I can imagine of already available tools supporting the coding. Furthermore what to do when the copy does not adhere to the syntax is left open. I welcome any suggestions: does regular grammars come in? Is YACC handy?

But, would I really like it? Honestly speaking, NO! Not at all. I just love so much the freedom \TeX provides. So why should I continue working in a direction that inhibits that? By the above I can protect myself, if needed,

against my own weaknesses. And that is that. However, the idea circulated around and out of the blue Andrew Dobrowolski from Arbor \TeX t commented the approach as practical unattainable. His idea for using \TeX 's stack mechanism looks promising. As an appendix I'm happy to add his opinions² with respect to the discussion.

Appendix (Andrew's comments)

From: arbortext!aed@sharkey.cc.umich.edu (Andrew Dobrowolski)

Answered 03/11 17:33 by JPC @ADMIN (see #10)

When discussions of using \TeX to enforce SGML like content models arise I think it is best to make the following clear right from the beginning:

Full SGML context checking by \TeX is a practical impossibility because \TeX lacks the necessary data structures to keep track of arbitrarily complex contexts.

This is not to say that some ingenious \TeX perts cannot emulate these structures using macros, but that the resulting code would run slower than my grandmother.

To begin with, we must admit the concession that \TeX is not going to read the SGML dtd, that is the SGML declarations which define the allowable document structures and their content models. Any change to the dtd would mean custom work by the \TeX programmer.

For a practical solution we will need to restrict the allowable content models that we are trying to enforce. At bare minimum we would have to allow the following:

- Comma groups of unrestricted length. That is a content model of the form (a, b, c, ...), meaning that this element must contain the element a, followed by the element b, and so on.
- Or groups of unrestricted length. That is a content model of the form (a|b|c|...), meaning that this element must contain one and only one of the elements a or b or c or ...
- The optional modifier "?", meaning that the preceding element need not appear. A paper may have an optional sub-title for example.
- The zero or more modifier "*" meaning that the preceding element may appear zero or more times. An example would be the paragraphs following a chapter title and preceding the first section.

To begin with we can exclude the less frequently used "&" groups, the one or more modifier "+" and the esoteric inclusions and exclusions. We can even restrict ourselves from mixing comma and or groups within one element's content model.

With these restrictions it would not be so difficult to get some limited context checking to work, assuming good

²L \TeX edited, adapted to two column format and some use of quoting for emphasizing important ideas.

faith on the part of the user. But I think there is more work to it than first meets the eye. To begin with, if we build context checking into the elements themselves, then we run into the problem that an element may appear in any number of structures (such as a title that may appear in the structures chapter, section, figure, table, extract, etc.) Context checking code for each of these situations would be different, and so the definitions of the most commonly used elements would be the most complex. This is a definite burden on performance. To avoid the burden the TeX programmer could redefine the title (using \let) for every context in which it is allowed.

But this is not yet enough to solve the content checking problem. An element may appear two or more times within the same model. For example in following model for the content of the element A, B appears twice: <!ELEMENT A - - (B, C, B, D)>. It needs to check that it is followed by a C in the first instance and by a D in the second. Once again, either B gets more complicated or C redefines B using \let. This gets very tricky if C is optional: <!ELEMENT A - - (B, C?, B, D)>. Which element redefines B if C is missing? If the first B then what if it is also optional?

Further complications appear when an element is allowed to contain itself such as when an emphasis tag is allowed to contain another emphasis tag. In the model <!ELEMENT B - - (A | B)> the end of the B element must perform different functions depending on whether it is the inner B or the outer B element. So even assuming simplified unmodifiable sgml content models, I do not think that having context built in to the elements themselves is a viable approach.

I would propose a different attack, one which takes advantage of TeX's stacking. I apologize if this gets too technical.

Define a macro data structure called a model. A model begins with a group type character (one of ",", "|" or "?" or "*") followed by one or more brace delimited groups. Each of these groups contains either a string representing an element name or another model. The topmost model ends with a period. The model would allow for comma groups, 'or' groups, the optional modifiers "?" and the zero or more modifiers "*". For example the model for "A" above may be defined:

```
\def\modelA{ , {B} { ? {C} } {B} {D} . }
```

where the definition begins with a (possibly active) character giving the group type (a comma group here) followed by the members of that group. The two group types are "," and "|". While the modifiers are "?" and "*". The final period is used to indicate the end of the model.

The current model is maintained in a local macro called \cmodel. As we work through the structures contained in A, the current model is modified to reflect what is still to be expected. \cmodel will take on values equivalent to the following definitions:

```
\def\cmodel{ , {B} { ? {C} } {B} {D} . }
\def\cmodel{ , { ? {C} } {B} {D} . }
\def\cmodel{ , {B} {D} . }
\def\cmodel{ , {D} . }
\def\cmodel{ , . }
```

Each of the structures within the "A" structure would first check themselves for context and report any context error as required. They would do this by calling a macro \ccheck that takes one argument, the name of the calling structure. So "B" would call \ccheck{B}, which in turn would examine \cmodel. It is possible that \cmodel will be modified by this call:

Body (replacement text) \cmodel

BEFORE \ccheck{B}	AFTER \ccheck{B}
{B}{ ? {C} } {B} {D} .	{ ? {C} } {B} {D} .
{ ? {C} } {B} {D} .	{D} .
{B} {D} .	{D} .
{D} .	{D} .
..	..

The last two calls would also cause error messages along the lines:

Element B is out of context here, expecting D. I will treat the B as an inclusion.

After B's context check operation is over, it starts a new TeX group and redefines \cmodel to be \modelB. The old \cmodel is still on TeX's save stack and will pop back when element B is finished.

It is easy to see how this can be extended to allow for more complex models, including the nesting of comma and or groups and the addition of and groups.

The beauty of the idea is that the SGML dtd no longer becomes part of the definitions of the macros that make up the document.

Its effect is only seen in the beginning of the macro file in the definitions of \modelA, \modelB, etc. I will not go into any more details here, since other projects are pressing for my attention, except to say that although I have not implemented such macros, they should not take more than a few hard days of work. If someone attempts to do this, I would not mind hearing about it at TUG (or sooner).

Andrew Dobrowolski
ArborTeXt
aed@arbortext.com

BIJLAGE P**Gebruik van T_EX binnen het EGD¹****H. Brouwer, J.A. Jager, P. Sader**

Maart 1991

Tijdens de presentatie van de EGD (Energiebedrijf voor Groningen en Drente N.V. te Groningen) zullen de volgende punten aan de orde komen:

- Conversie van tabellen binnen Quatro naar T_EX door J.A. Jager
- Het gebruik van macro's binnen het EGD door P. Sader.

Conversie van tabellen binnen Quatro naar T_EX

Bij de afdeling IV binnen het EGD bestaat de planning momenteel uit meerjaren- en jaarplannen voor zowel de investeringen als de exploitatiekosten.

In het meerjarenplan staat een overzicht van de kostensoorten en de bijbehorende uren en bedragen voor de komende 6 jaren. De splitsing in investeringen en exploitatiekosten is noodzakelijk voor de administratie. Het nadeel hiervan is dat één project op twee plaatsen in het verslag kan staan.

Het is dus de bedoeling om te komen tot een verzameling van tabellen met daarin de overzichten van zowel

de investeringen als de exploitatiekosten. Het geheel moet dan eenvoudig te splitsen zijn in 2 afzonderlijke verslagen, één voor investering en één voor exploitatie. Tevens moet het geheel opvraagbaar zijn in zowel een spreadsheet (Quatro) als een tekstverwerker (T_EX). Met behulp van eigen gedefinieerde macro's binnen Quatro wordt er nu T_EX code gegenereerd voor het weergeven van de tabellen.

Het gebruik van macro's binnen het EGD

Er wordt op dit moment binnen het EGD op kleine schaal gebruik gemaakt van T_EX. Het is de bedoeling dat dit in de toekomst gaat veranderen.

Om T_EX voor een groter publiek bruikbaar te maken, is er een set macro's ontworpen die het gebruik van T_EX vereenvoudigen. De gebruiker hoeft dan alleen kennis te hebben van de macro's en niet van de interne, relatief ingewikkelde, commando's van T_EX.

Een tweede reden is het standaardiseren van de documenten die door het EGD gemaakt worden. Voorbeelden hiervan zijn het gebruik van dezelfde lettertypes en structuren, zoals kolommen en lijsten.

¹ Presentatie te geven tijdens de NTG vergadering op 2 mei 1991

BIJLAGE Q**Gebruik van T_EX en L^AT_EX op het CAWCS**¹**Leo en Marion van Geest**

Februari 1991

*Wij zijn niet geïnteresseerd in T_EX
en L^AT_EX!*

1 Inleiding

Het CAWCS is het Centrum voor Automatisering van Wapen en Commando Systemen der Koninklijke Marine. Een marinebedrijf dat de operationele software ontwikkelt welke draait aan boord van boven- en onderwaterschepen en vliegtuigen. Daarnaast worden voor dezelfde platformen trainers gemaakt waarop bemanningen aan de wal onderricht krijgen.

Het CAWCS heeft ongeveer 150 medewerkers waarvan 100 zich daadwerkelijk met software ontwikkeling bezighouden. De overigen behartigen de operationele kant van de systemen of bieden administratieve en technische ondersteuning.

De auteurs zijn leden van de in totaal uit 5 personen bestaande sectie Support van de afdeling Systeem Ontwikkeling. De taak van deze sectie is het instant houden en onderhouden van de ontwikkelomgeving, ondersteuning van gebruikers, het ontwikkelen van nieuwe en onderhoud aan bestaande tools, de evaluatie, installatie en in gebruik name van software producten van derden etc.

2 Ontwikkelomgeving

De omgeving waarbinnen de software ontwikkeling plaats vindt bestaat uit een cluster van 6 VAX computers waaraan 19 VAX werkstations gekoppeld zijn, allen met een VMS operating systeem, alsmede enkele militaire computers. Gebruikers hebben de beschikking over VT200 en VT320 terminals, een aantal LN03 laser printers en enkele PostScript printers.

De voornaamste software pakketten die gebruikt worden bestaan uit Teamwork voor analyse en design, Oracle data base management systeem, Ada als programmeertaal voor de VAX computers, ondersteund door de complete toolset van DEC (VAXset) en Mars/Mascot met RTL/2 voor de militaire computers.

En, het zal U niet verbazen, T_EX en L^AT_EX met een heleboel gerelateerde tools.

3 T_EX en L^AT_EX binnen het CAWCS

T_EX, L^AT_EX en de gerelateerde tools worden binnen de afdeling Systeem Ontwikkeling van het CAWCS gebruikt voor het genereren van allerhande documentatie. De nadruk ligt hierbij op twee aspecten: L^AT_EX en gebruik.

Software productie is waar het om draait en daarbij behoort een aanzienlijke hoeveelheid documentatie. L^AT_EX wordt gezien als een tool waarmee het produceren van documentatie vereenvoudigd wordt terwijl de kwaliteit verbetert. Het werkt productie verhogend. T_EX daarentegen wordt door velen als 'te moeilijk voor het dagelijks gebruik' ervaren.

Binnen het CAWCS zijn slechts enkele T_EX kenners aanwezig. Samen met de medewerkers van de sectie Support, zorgen zij voor uitbreiding van de L^AT_EX omgeving. Alleen uitbreiding, want naast kwaliteit is continuïteit één van de belangrijkste aspecten binnen een productie omgeving. Wat vandaag kan moet morgen ook kunnen. Het mag wel sneller, mooier, uitgebreider maar niet anders.

4 Realisatie

Alle T_EX gerelateerde software betreft het CAWCS uit de DECUS T_EX collection. De sectie Support is verantwoordelijk voor kwaliteit en continuïteit en selecteert vooraf een subset uit de aangeboden styles, tools, etc. afgestemd op de ontwikkelomgeving en de gebruikerswensen.

In verband met toekomstige upgrades, wordt zoveel mogelijk gekozen voor standaard producten. Wijzigingen beperken zich tot het corrigeren van eventuele fouten, indien mogelijk, kleine aanpassingen aan styles en inpassing in de ontwikkelomgeving, gebruikers interface, opslag structuren etc. De meeste aangeboden styles en opties worden zonder meer overgenomen.

¹ Presentatie te geven tijdens de NTG vergadering op 2 mei 1991

Presentatie aan de gebruikers geschiedt in de vorm van manual pages en users reference manuals, verzameld in een zogenaamd Support Manual. In dit manual wordt alle algemene CAWCS software beschreven en de hele L^AT_EX omgeving is hierin ook ondergebracht. Het Support Manual wordt overigens ook geproduceerd met L^AT_EX. Hiervoor zijn speciale styles ontworpen, zowel voor de manual pages als voor de reference manuals.

5 De Selectie

Op het CAWCS worden naast L^AT_EX en T_EX de volgende tools in meer of mindere mate gebruikt:

- T_EXachtigen
 - BibT_EX
 - SliT_EX
 - GloT_EX
 - IdxT_EX
- DVI verwerking
 - DVI2LN3 (conversie naar LN03 code)
 - DVI2ALW (conversie naar PostScript)
 - CRUDETYPE (conversie naar text file)
 - DVI2TTY (preview op text terminal)
 - XDVI (preview op VAX station)
 - DVITOVDU (preview op verschillende terminals)
 - DVITYPE (DVI leesbaar maken)
- Algemeen
 - SPELL (spelling checker met T_EX know-how)
 - LSEDIT environment en section files (ter ondersteuning van het intikken t.b.v. L^AT_EX, BibT_EX en SliT_EX en met de mogelijkheid L^AT_EX te starten vanuit de editor en fouten te 'reviewen')
 - RNOtoT_EX (conversie Digital Standard Runoff naar T_EX)
 - TR2T_EX (conversie Troff naar T_EX)

6 De Uitbreidingen

De gekozen subset is op het CAWCS uitgebreid met styles en opties die aansluiten bij de wensen die ontstaan als men zich met systeem ontwerpen en programmeren bezighoudt. Bovendien dient men zich voor wat betreft de documentatie van de operationele software te houden aan bepaalde standaards.

Dit laatste heeft geleid tot twee document styles, QA-Doc en QADocLand, die met name de voorgeschreven layout ondersteunen. Beide styles zijn in principe gelijk, met dien verstande dat QADoc portrait mode en QADocLand landscape mode ondersteunt. De styles kenmerken zich door een grote bladvulling, een standaard pagestyle met daarin een kader en een vaste heading, de mogelijkheid een footer met daarin de document classificatie op nemen, paginanummering per hoofdstuk en een nesting van subsections tot maximaal 7 levels. De heading bevat een titel, subtitel, documentnummer, hoofdstuk- en pagina nummer, datum en auteur. Als gevolg hiervan

bleek het nodig de DVI converters voor LN03 en PostScript te voorzien van een vaste set qualifiers/opties en iets gewijzigde pre-amble files. Er is een omgeving gecreëerd waarbinnen L^AT_EX draait en waarin de gebruiker afhankelijk van de gebruikte documentstyle, de daarbij behorende DVI conversie kan kiezen.

Vanuit de hoek van de programmeurs is de wens gekomen om in de module design documentatie, verschillende data en controle structuren te kunnen opnemen. Hieruit zijn een aantal style opties voortgekomen die het mogelijk maken flow charts, Nassi-Schneidermann diagrammen, data structuren en bitpatronen in een document te definiëren.

Voor data structuren en bitpatronen is een L^AT_EX environment gemaakt waarbinnen bits, bytes en woorden gespecificeerd kunnen worden, voorzien van de nodige tekst, en waar een tekening uit komt van de gespecificeerde datastructuur.

Voor flow charts en Nassi-Schneidermann diagrammen is gekozen voor specificatie in de vorm van pseudo-code. Alle bekende structuren als if-then-else, while-do, case, repeat-until en de acties zijn gedefinieerd als commando's. Deze dienen gebruikt te worden binnen een specifiek commando waarin het totale diagram gedefinieerd wordt. Overigens zijn beide styles geschreven in T_EX.

Ten behoeve van alle bovenstaande, veel gebruikte, styles en opties is de Language Sensitive Editor omgeving uitgebreid met ondersteuning hiervoor. Deze ondersteuning is grotendeels ondergebracht in aparte files, die zonder problemen toegevoegd kunnen worden aan de uit de T_EX collection afkomstige ondersteuning.

Een andere ontwikkelde style is Manual, ten behoeve van het al eerder genoemde Support Manual. Hierin zijn commando's opgebracht die het mogelijk maken manual pages en users reference manuals te produceren. Een style optie Listing maakt het mogelijk source code op te nemen binnen deze style, wat vooral gebruikt wordt om specificaties van standaard (Ada) packages te documenteren.

Als laatste, de enige pure T_EX applicatie, Cards, een tool om eenvoudige visitekaartjes te maken.

7 Integratie en andere toepassingen

Behalve uitbreiding van de bestaande L^AT_EX omgeving, is er ook gezocht naar toepassingen waarbij L^AT_EX gecombineerd wordt met andere software ontwikkelings tools die bij het CAWCS in gebruik zijn.

Eén van de meest productieve inspanningen op dat gebied, heeft geleid tot een uitbreiding van de Document Production Interface (DPI) van Teamwork. Teamwork biedt de mogelijkheid om een document, de hoofdstuk indeling etc. te beschrijven in een Structured Chart. In de structured chart wordt verwezen naar het model dat het in ontwikkeling zijnde systeem beschrijft. Een

model is een verzameling entity relation diagrams, data flow diagrams, state transition diagrams, data dictionary entries etc. DPI biedt de mogelijkheid uit de structured chart en het model een standaard document te produceren in Interleaf, Scribe of VAXDocument formaat, maar bleek ook aanpasbaar en uitbreidbaar. En dat hebben we gedaan. DPI produceert nu ook L^AT_EX source files. De diagrammen worden geproduceerd in PostScript en ingevoegd in het document waarvan de structuur, de documentstyle etc. in L^AT_EX gedefinieerd is volgens de geldende normen.

Een ander voorbeeld is een tool, DataModel, waarmee het mogelijk is databases te ontwerpen. Met behulp van invulschermen en een databestand wordt de opbouw van de database vastgelegd of gewijzigd. De tool genereert twee dingen, de Oracle statements waarmee de database daadwerkelijk gecreëerd kan worden en een L^AT_EX source file die voldoet aan de standaard voor een zogenaamd Database Design Document met de complete benodigde beschrijving van diezelfde database.

8 Toekomst

Zoals uit het voorgaande moge blijken, zoekt het CAWCS het voornamelijk in de toepassing van L^AT_EX en in mindere mate T_EX. Ook in de toekomst zal de nadruk liggen op het uitbreiden van de toepassingsmogelijkheden. Behoud van wat er is vormt de belangrijkste overweging die speelt, bijvoorbeeld bij de upgrade naar een volgende versie van T_EX. Er zal een grondige evaluatie

aan voorafgaan.

Er zijn ook plannen en ideeën voor verdere ontwikkeling van tools en/of uitbreiding van bestaande. L^ASEDIT biedt sinds kort de mogelijkheid gebruik te maken van een Program Design Language en daaruit documenten te genereren volgens de Dod-STD-2167A, een documentatie norm waaraan ook het CAWCS moet voldoen. ASCII text en Runoff source zijn de mogelijke vormen van een dergelijk document, maar dat is uitbreidbaar en wat ons betreft is L^AT_EX daarvoor een uitstekende kandidaat.

Ook volledige ondersteuning, in de vorm van L^AT_EX templates in L^ASEDIT, voor de complete DoD-STD-2167A documentatie set behoort tot de toekomstplannen.

Een ander idee is een Ada source code formatter, waaruit niet alleen geformateerde (compileerbare) Ada source files komen maar ook L^AT_EX source files, waarmee dan een 'pretty printer' gerealiseerd is.

Verder zullen de toepassingsmogelijkheden van SGML en aanverwanten gezien worden.

9 Conclusie

Zoals gezegd: Wij zijn niet geïnteresseerd in T_EX en L^AT_EX,

Maar wel in wat het voor ons kan doen!

1 De document style QADoc

Dit document is tot stand gekomen door de volgende commando's :

```
\documentstyle[CMacs,nassi,nl]{QADoc}  
\docno{NTG-voorbeeld-001}  
\author{Leo en Marion van Geest}  
\date{28 februari 1991}  
\title{Voorbeeld van verschillend}  
\subtitle{CAWCS styles en opties}  
\class{Niet geclassificeerd}
```

De style QADoc verzorgt zowel de documentstyle als de pagestyle, standaard in 11 pts. De parameters voor de heading worden in de pre-amble gedefinieerd.

Behalve de layout valt bij deze style ook op dat elke sectie op een nieuwe bladzijde begint, subsecties daarentegen niet. Alle standaards als table of contents, list of figures, glossary, index etc. kunnen gebruikt worden. Elk van deze onderdelen komt in een aparte sectie.

2 De optie CMacs

De optie CMacs bevat onder andere de definities voor 'Bits' die gebruikt wordt om bitpatronen en data structuren te vertonen.

Een simpel voorbeeld is:

— VDU SWITCHED —

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	source=22 (MH1 active)															H1	
2	type=1							subtype=3							H2		
3	VDU number																ID

wat bereikt wordt met de commando's:

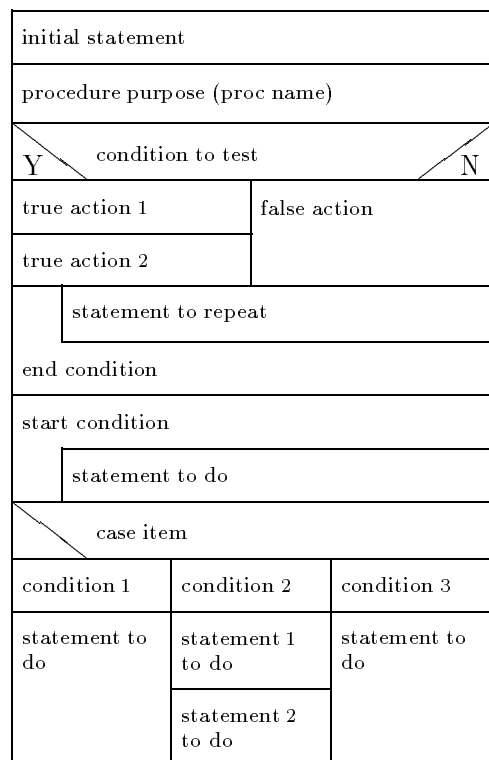
```
\beginbits{VDU SWITCHED}  
  \word{1}{source=22 (MH1 active)}{H1}  
  \bytes{2}{type=1}{subtype=3}{H2}  
  \word{3}{VDU number}{ID}  
\endbits
```

De parameter van `\beginbits` is de naam van de structuur. De parameters van `\word` en `\byte` geven de mogelijkheid om text voor, in en na de structuur-regel te specificeren.

3 De optie Nassi

De optie Nassi maakt het mogelijk Nassi-Schneidermann diagrammen te maken.
Het volgende diagram

Structure name — Structure purpose



wordt bereikt door de commando's:

```

\scriptsize
\STRUCT{Structure name}{Structure purpose}{%
  \ACTION{initial statement}%
  \PROC{proc name}{procedure purpose}%
  \IF{condition to test}%
  \THEN{%
    \ACTION{true action 1}%
    \ACTION{true action 2}%
  }%
  \ELSE{%
    \ACTION{false action}%
  }%
  \ENDIF%
  \REPEAT{%
    \ACTION{statement to repeat}%
  }%
  \UNTIL{end condition}%

```

Niet geclassificeerd

**Voorbeeld van verschillende
CAWCS styles en opties**

No. : NTG-voorbeeld-001
 Sect./sheet : 3/2
 Date : 28 februari 1991
 Author : Leo en Marion van Geest

```

\WHILE{start condition}{%
  \ACTION{statement to do}%
}%
\ENDWHILE%
\CASE{case item}{%
  \WHEN{condition 1}{%
    \ACTION{statement to do}%
  }%
  \WHEN{condition 2}{%
    \ACTION{statement 1 to do}%
    \ACTION{statement 2 to do}%
  }%
  \WHEN{condition 3}{%
    \ACTION{statement to do}%
  }%
}%
\ENDCASE%
}%
\normalsize
  
```

Door toevoeging van enige 'dimensie' commando's kan in breedte, en daarmee in hoogte gevarieerd worden. Met het commando `\nassewidth=\textwidth` wordt het volgende resultaat bereikt:

Structure name — Structure purpose

initial statement		
procedure purpose (proc name)		
Y / condition to test		N
true action 1	false action	
true action 2		
statement to repeat		
end condition		
start condition		
statement to do		
case item		
condition 1	condition 2	condition 3
statement to do	statement 1 to do	statement to do
	statement 2 to do	

Niet geclassificeerd

1 De optie Flow

De optie Flow maakt het mogelijk Flow charts te maken.

Het is niet mogelijk flow charts en Nassi-Schneidermann diagrammen samen te gebruiken in één document. Voor beiden wordt dezelfde pseudo-code gebruikt. Het volgende flow chart is gerealiseerd door vrijwel dezelfde commando's als die welke in het eerste voorbeeld document voor het Nassi-Schneidermann diagram gebruikt zijn.

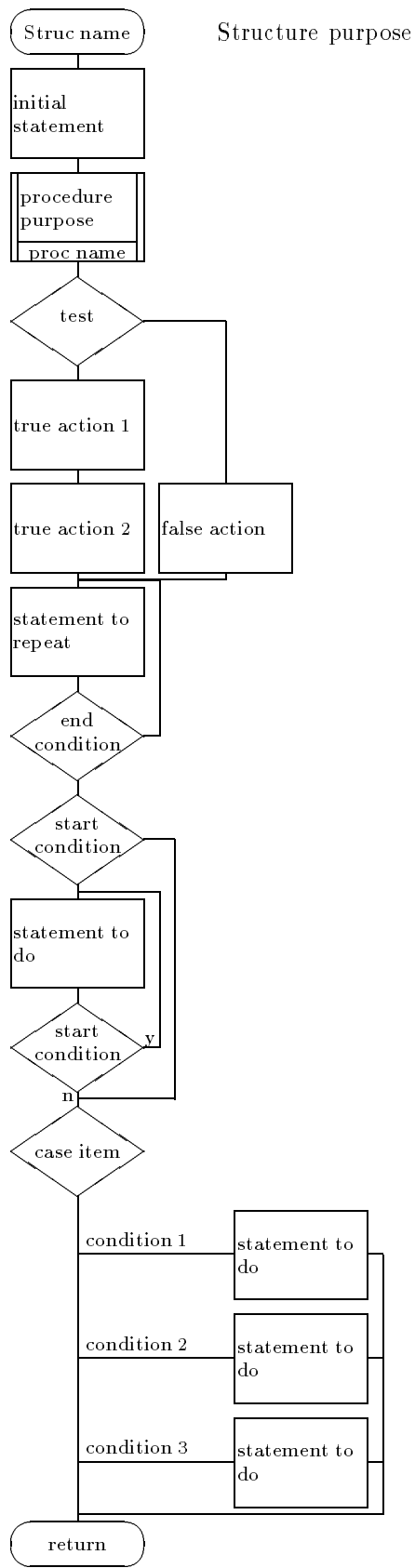
De commando's:

```
\scriptsize
\STRUCT{Struc name}{Structure purpose}{%
  \ACTION{initial statement}%
  \PROC{proc name}{procedure purpose}%
  \IF{test}%
  \THEN{%
    \ACTION{true action 1}%
    \ACTION{true action 2}%
  }%
  \ELSE{%
    \ACTION{false action}%
  }%
\ENDIF%
\REPEAT{%
  \ACTION{statement to repeat}%
}%
\UNTIL{end condition}%
\WHILE{start condition}{%
  \ACTION{statement to do}%
}%
\ENDWHILE%
\CASE{case item}{%
  \WHEN{condition 1}{%
    \ACTION{statement to do}%
  }%
  \WHEN{condition 2}{%
    \ACTION{statement to do}%
  }%
  \WHEN{condition 3}{%
    \ACTION{statement to do}%
  }%
}%
\ENDCASE%
}%
\normalsize
```

zorgen voor het volgende diagram

**Voorbeeld van verschillende
CAWCS styles en opties**

No. : NTG-voorbeeld-002
Sect./sheet : 1/2
Date : 28 februari 1991
Author : Leo en Marion van Geest



Niet geclassificeerd

BIJLAGE R

PROOF

Math into BLUes¹**Kees van der Laan**Hunzeweg 57, 9893PB
Garnwerd, The Netherlands
+31 5941 1525

March, 1991

Abstract

\TeX ing mathscripts is not simply typing. Math has to be translated into \TeX commands. First the motivation for this work is given. Next traditional math page make-up is summarized along with the macroscopic math \TeX commands. After answering ‘Why \TeX ing mathscripts is difficult?’ an anthology of \TeX falls and their antidotes is discussed. At the end suggestions are given in order to lessen the difficulties.

Prelude

My assistance was called for in \TeX ing a mathscript. Part of the mathscript was typed, contained \TeX commands, but . . . did not pass the \TeX formatter. Inspection revealed it never could have. It occurred to me that at least three typists had been involved, demonstrating the use of \LaTeX , \AMSTeX and macros from other sources, all mixed up. Furthermore, the \TeX script showed various \TeX falls. I like to define these as: correct \TeX ing not yielding the required or customary lay-out. Also the pseudo-guru involvement was felt, which I like to define as too complicated use of \TeX , inhibiting intelligibility of the \TeX script, with a wink to TB373

‘Always remember, however, that there’s usually a simpler and better way to do something than the first way that pops into your head.’

Not only did I look over the shoulder of a typist, I also inspected a math book, \TeX ed by a mathematician, Temme(1990). The book looks good. Examples are taken from it in order to show other ways of \TeX ing. A matter of taste?

In the sequel attention is paid to

- Traditional math page make-up.
- What makes \TeX ing mathscripts difficult?
- An anthology of \TeX falls² with antidotes.
- What ought to be done to lessen the difficulties.

For you and me

Most, if not all, (math) \TeX falls have been envisioned by the grand wizard himself and references to those or related issues are indicated by TB (The \TeX book) followed by page or exercise number.

Mathscript denotes a mathematics manuscript. \TeX script denotes a \TeX formatted compuscript, especially the one my assistance was asked for. \TeX nigma is a computer system with \TeX installed. \TeX knowledge means knowledge of \TeX . \TeX ist is a \TeX typist. BLUe is \TeX ’s unwary B.L. User. The math book denotes Temme(1990).

1 Math page make-up

Swanson(1986) is a good source for traditional math mark-up. In publications math is either part of the running text or displayed. In displays ‘indentation’ on all sides is on, and formulae are sometimes aligned, for example at the ‘=’-symbol.

\TeX requires math within text to be surrounded by ‘\$’-s, $\$<math>\$$. Displayed math is tagged by ‘ $\$>$ ’-s, $\$><math>\$>$. For the general multi-line display, plain \TeX provides the macro $\backslash displaylines$, TB194, 362, and for aligned formulae the macro $\backslash equalign$, TB190, 362. By default displays are centered. That is all for \TeX ing math, from an outer level point of view. Problems?

¹To be presented at Dedham91, TUG conference 1991, Paris91, Euro \TeX conference 1991 —© 1991, \TeX Users Group

²The \TeX falls treated are not specific to plain, \AMSTeX , nor \LaTeX . They illustrate basic pitfalls in \TeX ing math. Sources are: the inspected \TeX script, the math book and some I stumbled upon myself. \LaTeX is rather superficial with respect to math. Formula classes are not even mentioned! Dead wrong, but understandable from the viewpoint of descriptive mark-up.

Example (*Pascal triangle*)

```

          1
         1 1
        1 2 1
       1 3 3 1
      . . . . .
    
```

is obtained via

```

$$$$\displaylines{1\cr
  1\quad1\cr
  1\quad2\quad1\cr
  1\quad3\quad3\quad1\cr
  \hbox to 7em{\$.dot$.hss
    $.dot$.hss$.dot$.hss
    $.dot$.hss$.dot$.hss}
}$$$$
    
```

The example demonstrates two levels of formatting math: the inner level where the triangle has to be defined unambiguously —here in detail TeX commands, and the outer level where this triangle is positioned within the context —here \$\$-signs meaning displayed, and subject to the style of the publication series.

By phone one would say: Pascal’s triangle, you know, 1, with below it 1 and 1, and there below 1, 2, and 1, and there below 1, 3, 3, 1, etc. All centered. For formatting more precise information is needed than for talking math by phone, in order to eliminate ambiguity. A computer-based formatting system is not yet that intelligent.

Right or left formulae numbers can be provided after the tag \eqno, respectively \leqno, TB187, 362. Individual lines in a multi-line display can be numbered. Therefore the macro \eqalignno, respectively \leqalignno, is provided, TB192, 362.

In summary, all plain TeX’s math page make-up macros are demonstrated in the following templates.

$$\sin 2x = 2 \sin x \cos x \quad (\text{TB186})$$

$$F(z) = a_0 + \frac{a_1}{z} + \frac{a_2}{z^2} + \dots + \frac{a_{n-1}}{z^{n-1}} + R_n(z),$$

$$n = 0, 1, 2, \dots,$$

$$F(z) \sim \sum_{n=0}^{\infty} a_n z^{-n}, \quad z \rightarrow \infty \quad (\text{TB ex19.16})$$

$$\begin{aligned} \cos 2x &= 2 \cos^2 x - 1 \\ &= \cos^2 x - \sin^2 x \end{aligned} \quad (\text{TB193})$$

$$\begin{aligned} \cosh 2x &= 2 \cosh^2 x - 1 \\ &= \cosh^2 x + \sinh^2 x \end{aligned} \quad (\text{TB192})$$

obtained via

```

$$$$\sin2x=2\sin x\, \cos x
  \eqno({\rm TB186})$$$$
$$$$\displaylines{F(z)=
a_0+{a_1\over z}+{a_2\over z^2}+\cdots
+{a_{n-1}\over z^{n-1}}+R_n(z),\cr
    
```

```

\hfill n=0,1,2,\dots\,,\cr
\hfill F(z)\sim\sum_{n=0}^{\infty} a_n z^{-n},
\quad z\to\infty\quad\quad\quad\hfill
\llap{(TB ex19.16)}\cr}$$$$
$$$$\eqalign{\cos2x&=2\cos^2x-1\cr
&=\cos^2x-\sin^2x\cr
\eqno({\rm TB193})}$$$$
$$$$\eqalignno{\cosh2x&=2\cosh^2x-1({\rm TB192})\cr
&=\cosh^2x+\sinh^2x\cr}$$$$
    
```

Remark. It was difficult to get the example with labeled \eqalign right in two-column format. It would left justify because of insufficient space left by the big label. Removing the glue ‘\,’, before the \vcenter in the body of \eqalign forced TeX to center the formula, see TB189.

One can also use the general \halign macro. For example from TB ex22.9 we have

$$10w + 3x + 3y + 18z = 1, \quad (9)$$

$$6w - 17x - 5z = 2, \quad (10)$$

obtained via

```

$$$$\openup1\jot\tabskip=0pt plus1fil
\halign to\displaywidth{\tabskip=0pt
$\hfil#$$\hfil{#}\cr
$\hfil#$$\hfil{#}\cr
$\hfil#$$\hfil{#}\cr
$\hfil#$$\hfil{#}\cr
$\hfil#$$\hfil{#}\hfil$\tabskip=0pt plus1fil&
\llap{#}\tabskip=0pt\cr
10w&+& 3x&+&3y&+&18z&=&1,& (9)\cr
6w&-&17x& & &-& 5z&=&2,& (10)\cr}$$$$
    
```

I consider \cases, \begin{pmatrix} and \overbrace, respectively \underbrace, parts of formulae. For example: x is called an eigenvector with eigenvalue λ of the matrix

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

if Ax = λx, TB177. Obtained via

```

$$$$A=\pmatrix{
  a_{11}&a_{12}&\ldots&a_{1n}\cr
  a_{21}&a_{22}&\ldots&a_{2n}\cr
  \vdots&\vdots&\ddots&\vdots\cr
  a_{n1}&a_{n2}&\ldots&a_{nn}\cr}$$$$
    
```

2 Am I blue?

This section should have been filled with impressive and dazzling examples like those that go along TeX-product ads. Because of space restrictions they have been omitted.

3 What is wrong Doc?

Mathscripts differ from \TeX scripts. The source

```
$$x=1+\left(\frac{y^2}{k+1}\right)^{1/3}.$$
```

looks different from³

$$x = 1 + \left(\frac{y^2}{k+1} \right)^{1/3}.$$

Because of this discrepancy, the problem is how to get a correct \TeX script, starting from just a mathscript. The mathscript has to be translated into a correct \TeX script. This is difficult because of the complexity of math typesetting, and the inherited complexity of \TeX , if not of the bewildering variety of \TeX -based products.⁴

First, one has to find the appropriate format command, from nearly a thousand.⁵ In the TB the following is devoted to math formatting: chapters 16(11p), 17(21p), 18(23p), 19(14p), 22(242, ex22.9/11), 24(up to 281, 15p), 26(5p); appendices A (answers to exercises), B(6p, macros), F(13p), G(7p). Add to these the required general \TeX knowledge of how to use \TeX for non-complex documents, of how to use \TeX for general page make-up, of how to format tabular material (matrices, commutative diagrams), of how to handle output routines, of how to use non-default fonts, and nobody would consider \TeX to be trivial.⁶

Second, contents and context dependent extras have to be added, now and then, as demonstrated in this paper.

Third, once the \TeX language is mastered, the difficulty remains to locate and correct errors. Misconceptions as well as typos.⁷ So add to the above TB chapter 27, just for completeness.

Fourth, the (La) \TeX bugs and \LaTeX 's inconsistency. Once you have coped with all mentioned above, you are nevertheless thrown back by true bugs. I was trapped by \LaTeX 's quote environment when I tried for the opening quote to hang out. It did not work, even not after inserting `\null`.

Spivak(1986), has dealt with \TeX ing math in his delightful book, but alas it is not a proper extension. My attitude is to look for what is needed and to extend plain in a compatible way, keeping overhead as small as possible. Plain provides enough \TeX falls already.

4 The bad news

I like to start with mentioning the nasty small white space on a new line after a heading. This creature can be killed

by providing a comment symbol `—%`direct after the heading command. Just a warming-up for the unwary.⁸

The sequel in this section started as a list of pitfalls. It grew out into a discussion with antidotes. If readability for BLUE is reduced below par, I pitfailed.

4.1 Too many.

The too many pitfall is a serious one. It occurs when many incompatible products are used, which are partly, or not at all, understood.

In the typing project, \TeX ed chapters showed different approaches. $\mathcal{AMS}\text{-}\TeX$ was used in one, \LaTeX in another etc. This demonstrated involvement of several typists and the lack of a common approach. The document did not run either, showing that \TeX ing is one thing and getting it correct—if not alone for those braces—is quite another. This holds especially for typists not familiar with programming. Apart from the above, \TeX ing was done inconsistently. $\mathcal{AMS}\text{-}\TeX$ was used for some math symbols not available in plain, like \gtrsim . Commands like `\frac`, and `\overset` were used along with their plain functional equivalents. Obviously one typist was $\mathcal{AMS}\text{-}\TeX$ oriented, while others were not.

In conclusion. The \TeX script was far from correct, suffered from leaning upon too many tools, and for the rest was full of horrible \TeX falls. The math book didn't suffer from this \TeX fall, just plain, and an extra symbol or two.

I was trapped when preparing this paper. This paper uses `LTUGproc.sty`, and therefore \LaTeX . In \LaTeX `\eqalign` etc. are not available, so I defined them. But, ... I did not think of redefining `\centering`, which has another meaning within \LaTeX than within plain \TeX . As a consequence `\eqalignno` went wrong without saying so. It just did!

Another \TeX fall was that `\eqalign` did not center in two-columns format when `\eqno` was used as well! I had to inactivate the first glue item `'\,'` of `\eqalign`, see for explanation TB189. Weird.

I started with `LTUGproc.sty` because my contribution for Cork90 was done by it, and I thought of needing the picture environment. For the NTG MAPS PROOF version, as well as for the GUTenberg cahiers reprint, I will stay with \LaTeX . The final version will be in plain, of course.

Another failure was when using the quote environment with the first quote hanging out. That did not work, and so I fell back upon `\midinsert`,

³Note that the kind of parentheses and the kind of division notation have to be specified as well.

⁴In the sequel we restrict ourselves to plain \TeX , and assume that no fancy, friendly WYSIWYG user-interface is available.

⁵Cheswick(1990) has provided a KWIC with all the \TeX and \LaTeX commands. This is handy when in doubt whether a command is already in use.

⁶Beeton(1990) states that it was the intend of the $\mathcal{AMS}\text{-}\TeX$ -project to 'simplify input of complex mathematical expressions.'

⁷The \TeX ist task has been silently increased by the parsing and correcting of the \TeX script in order to provide proofs.

⁸This is overlooked in the Dutch course book on \LaTeX , and also in the Dutch 'brief' style, where the addressee label on the sequel page headings is preceded by that white space.

`\, \cdot \, .`

Interesting are expressions in exponents or indices. The math book contained

$$e^{-z \sinh t + \nu t}$$

which does not look nice because of suppression of space around the operator. Introduce explicit thinspaces, before and after the binary operator, or use parenthesis around the argument of the function.

In the \TeX script I encountered among others

```
\wit
$$|t| \quad < \quad | \quad x \quad -
  \quad (x + 1)^{\frac{1}{2}}
  \quad (x - 1)^{\frac{1}{2}} \quad |, $$
\wit
```

Spacing between formulae was not understood and done inconsistently. At many, many places unnecessary extra white space was ordered for; hundreds of `\, . \quad`, respectively `\qqquad` were abused.

On the use of `\(\) \quad`, I can best quote TB166.

‘The traditional hot-metal technology for printing has led to some ingrained standards for situations like this, based on what printers fall a ‘quad’ of space. Since these standards seem to work well in practice, \TeX makes it easy for you to continue the tradition: When you type ‘`\quad`’ in plain \TeX format, you get a printer’s quad of space in the horizontal direction. Similarly, ‘`\qqquad`’ gives you a double quad (twice as much); this is the normal spacing for the F_n example below.’

A little further on the page the reader’s attention is drawn to the different approach needed in alternating math and text in a paragraph. Just alternate, as simple as that.

$$F_n = F_{n-1} + F_{n-2}, \text{ for } n \geq 2.$$

Consistency can be enhanced by defining document elements, and subsequently use the element via the name. For example the real part of z can be obtained in math mode via `\Re z`, once we have defined

```
\def\Re#1{\rm Re}\, #1}
```

In the math book this was implemented via `\{ \cal R \} \, #1`, which is handy especially when real parts of quantities are used in formulae. In the \TeX script I also encountered the following subtle examples, which after correction read

$$C_\nu^\lambda(-z) = \cos \pi \nu C_\nu^\lambda(z) - \sin \pi \nu D_\nu^\lambda(z),$$

where after the arguments of the trigonometric functions ‘`\, .`’ (extra space) had to be inserted. In the math book similar situations were circumvented via parentheses, `\cos(\pi \nu)C`, via `\cos(\pi \nu)C`, no extra space has to be inserted before the opening parenthesis and after the closing parenthesis, TB170.

¹⁰Why $| -x |$, and not just $|x|$? Furthermore, norm fences don’t belong to the openings respectively closings class.
¹¹Another occurrence of *harmful* braces is given by `\cnt={1}` and the like, yielding an error message.

4.5 Class unawareness.

Below several examples are provided which demonstrate the unawareness of mathematical characters belonging to one of eight classes, TB154.

4.5.1 Innocent braces.

The pitfall is that braces are not harmless but yield a formula of class 0 within math mode! Confer

`\$a+b\$, \$a{+}b\$` and `a+b` with results $a + b$, $a + b$ and $a + b$. The first `+` is of class binary and takes spacing according to the table on TB170, and in the second the `+` is reduced to class zero, and takes only `\mathsurround` spacing, defaulted in plain to Opt.

Similarly, TB171, shows

```
\$|-x|$, \$\left|-x\right|$,
and \$\lfloor-x\rfloor$
```

with results $| -x |$, $| -x |$, and $\lfloor -x \rfloor$.¹⁰ In the math book I found $\gamma^*(a, x)$, as well as $\gamma \star (a, x)$, do you see what went wrong?

\TeX perts swing around with braces, especially in alignments where empty formulae are to be used now and then. Whenever innocence shows up don’t believe it!¹¹ So, for \TeX ing math, BLUE must understand the various atom classes, TB158 etc.

4.5.2 Whoops.

What about this

```
\def\Inn{\raisebox{1pt}
{\hbox{\$ \in \$}}}
```

The concept of a binary operator was not accounted for, yielding wrong spacing. \TeX could not know that the raised `\in` had to be considered as an operator. It had been reduced to class ordinary, taking `\mathsurround` spacing.

In the math book I found

$$2\pi i \operatorname{Res}_{s=e^{i\pi}} f(s) = -2\pi i e^{i\pi z}.$$

\TeX ed via

```
2\pi i \, \, \operatorname{Res}_{s=e^{i\pi}} f(s)
=-2\pi i \, \, e^{i\pi z}.
```

I would prefer for the `\Res` operator (in display and in agreement with Swanson(1986))

$$\operatorname{Res}_{s=e^{i\pi}} f(s) = -e^{i\pi z}$$

via

```
\mathop{\operatorname{Res}}_{s=e^{i\pi}}
f(s)=-e^{i\pi z}
```

An example where spacing has to be suppressed is `<name>`, via `\{ < > name < > \}`. The relational operators are not used as such, and coerced into class zero, by the braces. The latter example is taken from the BNF-notation of programming languages, denoting meta-linguistic variables.

4.5.3 Just a comma.

The number 3,14 innocently formatted as $\$3,14\$,$ would yield 3, 14.

The correct formatting is $\$3\{ , \}14\$,$ TB134.

The comma belongs to the punctuation class of math symbols and the surrounding braces —making a subformula—reduce it to class ordinary, which doesn't take extra spacing.

Remark. As part of text, the number could have been obtained via 3,14, no \$'s around it. Context dependency remains difficult.

4.5.4 With dots.

A dot is in use for a (binary) multiplication operator and as a punctuation mark. Three dots in a row don't yield the ellipsis result. The formatting of the ellipsis is context dependent: at the axis of the formula, at the baseline, vertical, or diagonal.

Binary operator vs. punctuation mark.

A multiplication in mathematics can be denoted by: $a \times b, a \cdot b,$ and also implicitly by a thinspace $a b,$ which has to be marked explicitly.

Typists, and those used to the old typewriter, err by using 'x' for '\times,' and by using the punctuation dot '.' instead of '\cdot,' the binary multiplicator operator positioned centrally, and for the last notation just a space for '\,,' gobbled away by T_EX in good mood.¹²

The general issue is that the handwritten symbols must be recognized from the context, as a punctuation symbol, as an operator or ... as significant space!

Colons: is there a difficulty?

A colon as a punctuation symbol can be obtained via the '\colon' command, and as a relation symbol, via ':', TB134. Examples are

$$f: A \rightarrow B, \quad \{x : x > 5\}$$

obtained via

$\$f\colon A\to B,\quad \{x:x>5\}\$$

The math book used ':' throughout.

4.6 CMR.

Text in displays and standard function names take traditionally roman fonts, Swanson(1986, Table IV).

$\sinh x$ was T_EXed by $\$ \backslash \hbox{\sin } h x \$,$ demonstrating bad handwriting, and wrong T_EXing. The T_EXist was not familiar with the hyperbolic function names and therefore could not compensate for the bad handwriting, and also apparently lacked awareness of how to T_EX them properly. I also encountered

$\backslash \cos \backslash , \backslash \alpha$
 $h^{\backslash \lambda \backslash \nu} (z) \backslash , : = _2F_1(\dots)$
 $\backslash \hbox{\for Re } \backslash , z \backslash , > \backslash , 0$

Horrible!

¹²To this category of misuses I also reckon 1 vs. l, 0 vs. o.

¹³T_EX does adapt just (or \{.

4.7 Kameleons.

To the kameleon pitfall I reckon the situations where T_EX can't distil from the context the right sizing. T_EX provides facilities for automatically formatting the right size, given the context. T_EX provides for example the right-sized openings and closings for a matrix, when these are specified by $\backslash \left \dots,$ respectively $\backslash \right \dots$ ¹³ A T_EXfall occurs when the context does not prompt for any need of another size, while BLUe expects T_EX to do everything right.

Example (*Context dependent sizes*) Inspired on Spivak(1985, p55)

$\$ \$ \left| \left| \backslash \alpha \left(\backslash \sqrt{a} + \backslash \sqrt{b} \right) \right| \right|$
 $\backslash \leq \left| \backslash \alpha \right| .$
 $\left(\left| \backslash \sqrt{a} + \backslash \sqrt{b} \right| \right) . \$ \$$

with the result

$$\left\| \alpha (\sqrt{a} + \sqrt{b}) \right\| \leq |\alpha| \cdot (\left\| \sqrt{a} + \sqrt{b} \right\|).$$

Better T_EXing is

$\$ \$ \backslash \bigl \left| \left| \backslash \alpha \left(\backslash \sqrt{\mathstrut a} + \backslash \sqrt{\mathstrut b} \right) \right| \right| , \backslash \bigr \left| \right|$
 $\backslash \leq \left| \backslash \alpha \right| ,$
 $\backslash \bigl \left| \left| \backslash \sqrt{\mathstrut a} + \backslash \sqrt{\mathstrut b} \right| \right| \backslash , \backslash \bigr \left| \right| \$ \$$

with result

$$\left\| \alpha (\sqrt{a} + \sqrt{b}) \right\| \leq |\alpha| \left\| \sqrt{a} + \sqrt{b} \right\|$$

In this example norm fences are taken larger and all sqrt's must be told to have arguments of \mathstrut size. Ascender and descender invariance! Moreover the multiplication dot can better be replaced by a thinspace.

Another use of the vertical bar occurs in set notation, for example, TB ex18.22,

$$\{ x^3 \mid h(x) \in \{-1, 0, +1\} \}.$$

obtained via

$\$ \$ \backslash \bigl \left\{ \left| \left| x^3 \backslash \bigm | h(x) \backslash \text{in} \% \backslash \{-1, 0, +1\} \right| \right| \backslash , \backslash \bigr \left\} \right. \$ \$$

This not only demonstrates to use the correct size of the outer braces and the vertical bar, but also to be aware of the binary operator function of the vertical bar, defaulted with the appropriate spacing. Set notations in the math book had not been marked up via the use of $\backslash \mid |,$ nor by its variants. For nested parentheses the big etc. representations were not used. The old technique with square brackets for the outer parentheses was used. For example, $[\ln(z + 1)]^m.$

Note that it looks better to introduce some spacing along with the outer braces. I expect these kinds of issues to be handled inconsistently in a document of non-trivial size. If an author wants these kinds of results he has to indicate that in the mathscript.

4.8 Triads.

‘Three dots in a row,’ or generalized ellipsis, is heavily used in mathematics notation. For example

$$x_1 + \cdots + x_n, \quad x_1 x_2 \dots x_n$$

is obtained by using the `\cdots` and `\ldots` commands, respectively.

The general issue is not to use ‘...’, but to use the `\cdots`, respectively `\ldots` command.

Remark. When the ellipsis is followed by a punctuation dot a small extra space ‘\,’, has to be ordered for: `1+x+x^2+\cdots\,`. An ellipsis is often used in a fixed context, for example: for $i = 1, 2, \dots, n$. This can be obtained via

for $\$i=1\$, \sim 2, \$, \sim \$n\$$.

Such sentences are candidate for abbreviation into `\for in`, by use of

```
\def\for#1#2{
  for $\#1=1$,~2, $\ldots\,$,~$\#2$},
```

supporting consistency. It also reduces the number of keystrokes. Note that `\dots` is not substituted for `\ldots\,`, because `\,` is needed. The pitfall of mixing up `\dots` and `\ldots` use, is also circumvented by the use of the `\for` abbreviation.

The math book was inconsistent in using `\cdots`, respectively `\ldots`.

In order to facilitate the look up of the shorthands, Wichura(1990) has provided some macros yielding a table consisting of a math-writing-column and a corresponding T_EX-input column. A fancy tool, suited for typists, I presume. This is not enough in order to solve the typist’s problems. It might help, though. Education is needed and discipline has to be adhered to. What about a discipline of T_EXing?

4.8.1 Real life.

Other ‘dots’ are also in use: vertical, in matrices, TB177, and diagonal, TB177, ex18.45. As an example of diagonal dots consider, related to TB142,

$$1 + \Phi \frac{a_k}{b_k} \stackrel{\text{def}}{=} 1 + \frac{a_1}{b_1 + \frac{a_2}{b_2 + \dots + \frac{a_{n-1}}{b_{n-1} + \frac{a_n}{b_n}}}}$$

with (space saving) variant notations

$$= 1 + \frac{a_1}{|b_1} + \frac{a_2}{|b_2} + \dots + \frac{a_n}{|b_n}$$

$$= 1 + \frac{a_1}{b_1 + b_2 + \dots + b_n}$$

The above is obtained via

```
\def\cf{\mathop{\grkop \Phi}}
```

```
$$\ealignno{
1+\cf_{k=1}^n{a_k\over b_k}
&{\}\buildrel{\rm def}\over=
1+{a_1\over\displaystyle b_1+
{\strut a_2\over\strut
\vrule height3ex width0pt\relax
\displaystyle b_2 +
\lower2.0ex\hbox{\ddots},
\lower1.25ex\hbox{\$+
{\displaystyle a_{n-1}\over
\displaystyle b_{n-1}+
{\strut a_n\over
\displaystyle b_n}}}$}
}
}\cr
\noalign{\\noindent with variant notations}
&{\}\buildrel{\rm\phantom{def}}\over=
1+{a_1\,
\smash{\vrule depthlex}\vrule height2ex
\over\strut\vrule\,b_1}
+{a_2\,
\smash{\vrule depthlex}\vrule height2ex
\over\strut\vrule\,b_2}
+ \cdots
+{a_n\,
\smash{\vrule depthlex}\vrule height2ex
\over\strut\vrule\,b_n}\cr
%
&{\}\buildrel{\rm\phantom{def}}\over=
1+
{a_1\over\textstyle\strut
\vrule height2.5ex width0pt
b_1\,+\,},
{a_2\over\textstyle\strut
\vrule height2.5ex width0pt
b_2\,+\,},
\cdots
{a_n\over\textstyle\strut
\vrule height2.5ex width0pt
b_n}
\cr}%end\eqalignno
$$
```

The general issue is awareness of styles—display, text, script or scriptscript style—which is default where, and how to override the default, TB140 etc. Awareness of the difference between `\strut` and `\mathstrut` is needed as well. In the math book similar diagonal dots were used for denoting an infinite continued fraction. Swanson(1986) just provides an ellipsis. Her variant notations differ a little. Those given originate from Henrici. His Φ symbol is the absolute space saver for regular continued fractions.

Remark. The auxiliary symbol `\cf`, the Φ , must be made robust, such that it can be used with other styles as well, yielding suitable size. T_EX provides for this aim `\mathchoice`. For example `\cf` should have been defined as

```
\def\cf{\mathop{\mathchoice{\%
\grkop\Phi}{%Magnified
\Phi}{\Phi}{\Phi}}
}}
```



```

\lambda_0=-{\textstyle
\sin{\pi\nu\over2}\,
C^{\scriptscriptstyle\lambda\over
\scriptscriptstyle2}
_{\kern-1pt{\scriptscriptstyle\nu\over
\scriptscriptstyle2}}
} $$

```

In the math book I encountered the above notation, and incidentally $F(1/2, 1/2; 3/2; z^2)$. I also stumbled upon $\int^{\frac{1}{2}\pi}$ with later on the more usual $\int^{\pi/2}$. The latter is also recommended by Swanson(1986).

The general point is to kern and coerce the right style. Another example of where coercing the right style is needed, occurs when the summation symbol takes double lower limits. Explicit mentioning of `\scriptstyle` in both operands of the `\atop` command is needed, TB145.

Remark. D_EK(1985), mentions the use of a typographer's '1/2,' especially in recipes, which works better than a mathematician's '1/2'.

4.11 Various OOOOOO's.

Mathscript O's are overloaded: '∅,' the empty set, $f \circ g: x \mapsto f(g(x))$, composition, and the order symbols $o(h^2)$, and $O(h^2)$. Obtained via

```

\emptyset$,
f\circ g\colon
x\mapsto f\bigl(g(x)\bigr)$,
\mathcal{O}(h^2)$,
\mathcal{O}(h^2)$

```

We also have trigonometric and temperature degrees 30°, respectively °K, TB180. Another challenge is a notation for the zero vector, TB ex18.6.

4.12 Backslash penances.

Because of the special function of the backslash, people are in trouble when the symbol itself is wanted. In horizontal mode the backslash as such can be obtained by selecting the symbol from the `tt-font`, TB429, position '134 (decimal 92), via `\tt \char'134`. In math the backslash is used for the setminus (binary) operator and for denoting cosets, the latter takes no space.

Compare

$$A \setminus A = \emptyset \quad \text{and the cosets of } G \text{ by } H: G \setminus H.$$

T_EXed by use of `\setminus`, respectively `\backslash`, TB436. Unnecessary to say that the mathscript contained several setminus operations, while in the T_EXscript the `\backslash` was used throughout.

4.13 Over and over.

In TB ex17.3 BLUe is imprinted to treat a fraction as a subformula, id est, to use braces around `<formula-over-formula>`. A good habit to adhere to throughout. I was trapped when changing `\left(` and

`\right)` into `\bigl(`, respectively `\bigr)`. The former pair creates a subformula while the latter don't. Ahhhhhh, robustness!

Swanson(1986) advises to consider the use of slashes when saving space can be obtained with clarity of exposition preserved.

In `\buildrel`, TB437, `\over` is overloaded.

4.14 Too difficult.

Hypergeometric functions take sometimes 'matrices' as arguments. As stated in TB178, the use of `\(p)matrix` in the text of a paragraph yields too big results. $M_n(z) = {}_{n+1}F_n \left(\begin{matrix} k+a_0, k+a_1, \dots, k+a_n \\ k+c_1, \dots, k+c_n \end{matrix}; z \right)$ is obtained via

```

M_n(z)={}_{n+1}F_n
\bigl(\{k+a_0,\atop\phantom{kc_1}\}
\{k+a_1,\dots,k+a_n\atop
k+c_1,\dots,k+c_n\}
;z\bigr)$

```

Note the automatic centering 'on the axis' of the last argument. A fuzzy issue is what to do with empty arguments, especially when several `\atop`-s are used in a row. The general approach is to use `\mathstruts`-s. For two `\atop`-s the use of `\phantom` will yield aligned results, as demonstrated in the given example.

The late Luke used instead of ';' the '|' symbol, for example

$${}_pF_q \left(\begin{matrix} \alpha_p \\ \rho_q \end{matrix} \middle| z \right) = \frac{\Gamma(\rho_q)}{\Gamma(\alpha_p)} G_{p,q+1}^{1,p} \left(-z \middle| \begin{matrix} 1 - \alpha_p \\ 0, 1 - \rho_q \end{matrix} \right)$$

is obtained via

```

\mathcal{F}_p\mathcal{F}_q
\bigl(\backslash,\{\alpha_p\atop\rho_q\}
\mathpunct{\bigm|}\backslash,z
\bigr)=
\{\Gamma(\rho_q)\over
\Gamma(\alpha_p)\}\backslash,
G^{1,p}_{p,q+1}
\bigl(-z\backslash,\mathpunct{\bigm|}\backslash,
\{1-\alpha_p\atop0,1-\rho_q\}
\bigr)$$

```

Here the vertical bar is coerced into a punctuation symbol, with some extra spacing added. Note also the lack of spacing in the subscripts and superscripts, TB170.

The T_EXscript contained

```

\begin{dispeqs}
M_n(z):={}_{n+1}F_n
\left(\aligned
k+a_0,&&k+a_1,\dots,k+a_n;z\backslash
&&&k+c_1,\dots,k+c_n
\endaligned\right)
\tag1.2.55
\end{dispeqs}

```

not reflecting the centering of ' ; z.' Fortunately, I never needed to talk about hypergeometric functions by phone.

4.15 As simple as math can be.

Formula numbers are perhaps the most simple math elements of a math paper. Just numbers. This way of formula numbering suffers from the modification pitfall. The disadvantage of explicit numbering comes to light when copy is changed, involving modification of formula numbers. Authors circumvent retyping of the numbers by introducing suffixes a,b,c, . . . , when numbered formulae have to be inserted, and sacrifice strict sequencing when numbered formulae have to be omitted. Nowadays retyping can be circumvented by automatic numbering, jeopardizing the formula-number tie which some authors have developed. The better alternative is not thinking in number-formula ties, but in name-formula ties. How about that?

4.15.1 Automatism.

Plain does not provide for automatic numbering of formulae. One can easily define a new counter and write a macro with the function to advance the counter globally and provide the number, preceded eventually by chapter and section number with appropriate punctuation (Remind the mode you are in however, TB ex19.6, 19.7). In the example below we assume that chapter and section counters, `\cct`, respectively `\sct`, exist next to the formula counter `\fct`, and are handled appropriately when entering a chapter or section. For example, the numbering in

$$|x| = \begin{cases} x, & \text{if } x \geq 0 \\ -x, & \text{if } x < 0 \end{cases} \quad (2)$$

is \TeX ed by

```
\fct=1 \def\frmcnt{\global\advance\fct1
(\the\fct)}
$$|x|=\cases{
  x,& if $x\ge0$\cr
 -x,& if $x < 0$\cr}
\eqno\frmcnt$$
```

Note that the ‘second column’ is in horizontal (text) mode¹⁶, and that numbering of the formula is at the axis of the formula. The above approach can be used in `\eqalignno` as well. After the second &, insert again `\frmcnt`. For authors who cherish their habit to add a, b etc. to the number there is no problem. They can provide the counter with add-ons, for example the label (2a) is simply obtained via

```
$$|x|=\cases{
  x,& if $x\ge0$\cr
 -x,& if $x < 0$\cr}
\eqno(\the\fct{\rm a})$$
```

¹⁶While for number fields math mode is defaulted, I consider it an error-prone exception to have horizontal mode defaulted in the second column of cases. Not serious, though, it can be easily adapted.

¹⁷I hate trickery like `\ifnum0<0\csname . . . , which is in Pittman’s code. How long does it take to find the \TeX fall in \ifnum\cnt=0\else . . . \fi? The point is that the next symbol after a number will be expanded, in order to find out whether the number ended, TB208. So \else will be swallowed, adding the part after \else to the token list when the condition is true and omitting it when the condition is false. Innocent spaces!?!`

¹⁸Assuming the editor is \TeX intelligent, see Williams and Hall(1990).

¹⁹To reduce confusion, and support mnemotechnique, the discipline is prompted that each *formula* reference name begins with `for`.

Pittman(1988)¹⁷ and Nearing(1989) have provided macros for extending plain with automatic formula numbering (and symbolic referencing as well). Both did fail with respect to compatible *extension* of plain, although their macros can be easily adapted. I would unite Nearing’s `\eqnum` and `\eqalignnum` into for example the `\frmcnt` command, with the functionality as demonstrated above. `\frmcnt`, yielding the counter value, with the current lay-out embellishments, can be used as well after the `\(1)eqno` tag as after the second & of `\(1)eqalignno`. And why not along with `\displaylines`? Yes, you are right, that is compatibility!

What about the typist? Usually, the author provides the numbers as an integral part of the mathscript. I consider it easy to replace the number by the systematic call of `\frmcnt` at the place of the formula number. As simple as that, and can be done by one keystroke.¹⁸

The awareness that numbers should be typed after `\eqno`, or after the second & in `\eqalignno`, has to be applied anyhow.

4.16 You name it.

By symbolic referencing we mean linking a number to a name and to refer to the number via the name. This differs from automatic numbering because of the multitude of names, while automatic numbering use *one* counter (name). In so-called forward referencing the use of the name precedes the assignment of the number to the name. Linking a name to a number can be achieved via macro assignment. References made to the formula after this assignment can be done just via the call of the appropriate macro. That is the principle. The lay-out of the numbers has to be addressed somewhere, favorably in the format.

Example (*Link $\langle name \rangle$ to number*)

$$a^2 + b^2 = c^2 \quad (5)$$

is obtained, with `\forpyth` linked to (5), via

```
$$a^2+b^2=c^2\eqno\gdef\forpyth{(5)}
(5)$$
```

Subsequent references to this equation can be done via the ‘name’ `\forpyth`, with the number to be delivered in the format style of the context.¹⁹

An inconvenience is forward referencing. A suggestion how to handle this, is to print in the margin at the place

where the forward referencing was done a reminder that a forward reference has been made. One could print for example the $\langle name \rangle = ???$ (with $???$ the placeholder for the forward reference number). Sooner or later the numbers have to be filled in.

Example (*Forward referencing*) At this place we like to refer forward to the Pythagoras equation, ($???$), via $\backslash\text{forref}\{\backslash\text{forpyth}\}\{???\}$.

“forpyth
=(???)

$$a^2 + b^2 = c^2 \quad (1.2.3)$$

Above we displayed Pythagoras equation (3). Obtained via

```
\def\forref#1#2{
%#1 label name starting with \,
%#2 is number or ??? if unknown
%LaTeX's \marginpar is used.
  (#2)
\ifproof\marginpar{
  \vtop{\hbox{\string#1}
    \hbox{=(#2)}}}\fi
}%end forref marking}
$$a^2+b^2=c^2
\eqno%equation 3, chapter 1, section 2
  \xdef\forpyth{(3)}(1.2.3)$$
```

It is assumed that the author knows which name is linked to which equation, so that he can easily find the number and fills it in. Of course when the number is known, it can be typed in directly from the mathscript, but use $\backslash\text{forref}$ en type the number instead of the $???$. One never can tell.

4.17 Both ways.

Automatic counting and symbolic referencing can be combined. Assume that the running formula counter is called $\backslash\text{fct}$, and that the running chapter and section counters are called $\backslash\text{cct}$ and $\backslash\text{sct}$, respectively. The above example extends into

$$a^2 + b^2 = c^2 \quad (1.2.3)$$

obtained via

```
$$a^2+b^2=c^2
%hidden values: \fct, \cct, \sct
\eqno\global\advance\fct1
  \xdef\forpyth{(\the\fct)}
  (\the\cct.\the\sct.\the\fct)$$
```

In order to reduce the number of keystrokes the above functionality can be covered by a $\backslash\text{labf}$ ²⁰ command, appropriately defined. For example

```
\def\labf#1{
%links formula number to label
%#1 label: \langle name \rangle
%\fct is advanced and
```

²⁰I chose for $\backslash\text{labf}$ but a general $\backslash\text{lab}$ can be made which inspects an environment parameter in order to decide which counter has to be advanced.

²¹Note that in the display the chapter and section number are also printed, while a reference in the sequel text yields the number according to the appropriate format.

```
%via xdef assigned to \langle name \rangle
  \global\advance\fct1
  \xdef#1{(\the\fct)}
  (\the\cct.\the\sct.\the\fct)}
```

with the use

```
$$a^2+b^2=c^2
%hidden values: \fct, \cct, \sct
\eqno\labf\forpyth$$
```

References to the formula go via the name $\backslash\text{forpyth}$, and the lay-out defined via the format style of the publication is obtained.²¹ I chose to surround the number with parentheses. Fully automated symbolic reference systems are part of \LaTeX and \AMSTeX . Generally, use is made of an auxiliary file, which stores the numbers, assigned during the previous run. With forward referencing two runs are needed. Moreover, in those system one can also ask for the page number where the reference was made. Cross-referencing between document parts processed separately and independently, is also supported!

4.18 We want more.

Sometimes more alignment positions than the one position provided by $\backslash\text{eqalign}$, (or $\backslash(1)\text{eqalignno}$), are wanted. See for example TB242, where it is suggested to use $\backslash\text{eqalign}$ repeatedly within one display, and ex22.9 (equation 8 and 9 in this paper) for a general approach via $\backslash\text{halign}$. Repeated use of $\backslash\text{eqalign}$ suffers from non-guaranteed equal line distances over the various $\backslash\text{eqalign}$. In the math book I encountered

$$\begin{aligned} \cos(z \sin \theta) &= J_0(z) + 2 \sum_{n=1}^{\infty} J_{2n}(z) \cos 2n\theta \\ \sin(z \sin \theta) &= + 2 \sum_{n=1}^{\infty} J_{2n+1}(z) \sin(2n + 1)\theta \end{aligned}$$

which was not appropriately aligned, possibly because of not using the empty formula. Appropriate input

```
$$\eqalign{
\cos(z\sin\theta)={}&
&\&J_0(z)&+2\sum_{n=1}^{\infty}
&J_{2n}(z)\cos2n\theta\cr
\sin(z\sin\theta)={}&
&\&+2\sum_{n=1}^{\infty}
&J_{2n+1}(z)\sin(2n+1)\theta\cr
}$$
```

With only one alignment position the same result could have been obtained with $\backslash\text{phantom}$, and again the empty formula, $\{\}$.

4.18.1 Adaptation of `\eqalign`.

`Eqalign` can easily be adapted with respect to a variable number of alignment points. The number of alignment points does not have to be provided explicitly. No parameter is needed. The idea is to make use of the repetition facility for template elements in `\halign`; just double the appropriate `&`.

```
\catcode\@=11\@ letter
\def\eqalign#1{\, <---
    %Glue omitted in 2-columns env.
%Variable number of alignment points,
%determined dynamically, and
%automatically.
\vcenter{\openup1\jot\m@th
\ialign{\strut\hfil
    $\displaystyle{##}$&& %doubled
    $\displaystyle{\{ \}##}$\hfil\cr
    #1\cr}}\,
}% end flexible \eqalign
\catcode\@=12\@ other
```

With two alignment positions we can nicely format

$$(z^2 - 1)^\lambda = (z - 1)^\lambda(z + 1)^\lambda, \quad \operatorname{Re} z > 0, z \notin [0, 1] \\ = e^{\mp 2i\pi\lambda}(z - 1)^\lambda(z + 1)^\lambda, \\ \operatorname{Re} z < 0, \operatorname{Im} z \gtrless 0$$

via²²

```
$$\eqalign{(z^2-1)^\lambda
&=(z-1)^\lambda(z+1)^\lambda,
&\quad\operatorname{Re} z>0,\ z\notin[0,1]\cr
&=e^{\mp 2i\pi\lambda}
(z-1)^\lambda(z+1)^\lambda,
\hidewidth\cr
&\quad\operatorname{Re} z<0,\ \operatorname{Im} z\gtrless 0\cr
}$$
```

Note the use of `\hidewidth`. `\gtrless` will be discussed later. In the math book no alignment was done on occasion for function pairs, such as

$$P_\nu^{-\mu}(z) = \dots$$

and

$$Q_\nu^{-\mu}(z) = \dots$$

Spivak(1986), provides a general `\alignat#1` macro, where the parameter provides the required number of alignment positions.

4.18.2 Adaptation of `\(1)eqalignno`.

Pragmatically, one can use the general `\halign` command. One can also copy `\eqalignno`, extend the template and give the modified version an appropriate name. This is simple and serving its purpose. The more so because having more than one alignment position is the exception rather than the rule.

For a general poly-macro the number of alignment positions must be provided, because the last element of the alignment template is reserved for the formula number, and differs from the in-between ones. The automatic repetition mechanism can't be used. From the given value the template elements for the in-between alignments can be repeated, dynamically. The dynamical approach can be achieved via the use of `\aftergroup`. For the idea see the paradigm of defining a control sequence for `\n asteriks`, TB374.

Another possibility is selection of the appropriate code via a case-like construction.

4.19 Looooooooooooooooooooo.

With long formulae we have two situations. First when in-line formulae extend the line, and second when in displays formulae are too long.

For the first case \TeX breaks at customary points like: *after* binary operations and relations, TB195.

For the second case \TeX does not automatically break displayed formulae. The reason is that displayed math is too complex to automate line breaking, under the restriction to convey optimally the meaning of the formulae. The author knows best where to split a formula, and has to prompt \TeX and ipso facto the typist.

Authors should be aware of the typographic tradition to break in displays *before* binary operations and relations. For a summary of the rules see Swanson(1986, 3.3.4).

\TeX ing splitted formulae (in display) can be done via

1. Just use (two) consecutive displays.

This produces too much vertical white space and both equations are centered, which does not look good.

2. To use `\(1)eqalign(no)`.

The alignment position must be chosen. A good alignment choice is the `=` symbol, and to precede the second part with `&\qqquad{}`, the extras. You won't find the latter in the mathscript. Schematically,

```
\eqalign{
    <LHS>&=<RHS first part>\cr
    &\qqquad{ }<RHS sec part>\cr}
```

In the math book no 'extras' were inserted, just aligning at `=` and on the next line(s) with the `+/-`.

3. To use the free-format `\displaylines` command, see TB194.

4.19.1 Via consecutive displays.

The interdisplay width could be adapted. Because of simpler, self-contained and already available alternatives this is not further elaborated.

²²The above could have been obtained by using the phantom mechanism with alignment position chosen at `\Re`.

4.19.2 Via eqalign.

First, the typist has to be aware of the scope restrictions. For example `\left(` and `\right)`, can't be split, can't be used in isolation. They should be given within one scope, TB196. In `eqalign`(no) terms: they can't be separated by '&.' The non-context dependent variants (i.e. of fixed size), like `\Bigl(` respectively `\Bigr)`, can be split. Second, the line distance may vary unnoticed, because of the context.

Example (*Non-constant line distances*) The example does not deal with splitting a one-line formula into two parts, but is taken from the math book, modified to dramatize the effect.²³

$$\begin{aligned} E_0(x) &= 1, \\ E_1(x) &= x - \frac{1}{2}, \\ (E_2(x) &= x^2 - x)^2, \\ E_3(x) &= x^3 - \frac{3}{2}x^2 + \frac{1}{4}. \end{aligned}$$

is obtained via²⁴

```


$$\begin{aligned} E_0(x) &= 1, \\ E_1(x) &= x - \frac{1}{2}, \\ (E_2(x) &= x^2 - x)^2, \\ E_3(x) &= x^3 - \frac{3}{2}x^2 + \frac{1}{4}. \end{aligned}$$


```

Constant line distance can be obtained by the use of `\smash`, and editing at the proof phase.

Example (*Constant line distance*)

$$\begin{aligned} E_0(x) &= 1, \\ E_1(x) &= x - \frac{1}{2}, \\ (E_2(x) &= x^2 - x)^2, \\ E_3(x) &= x^3 - \frac{3}{2}x^2 + \frac{1}{4}, \end{aligned}$$

is obtained via

```


$$\begin{aligned} E_0(x) &= 1, \\ E_1(x) &= x - \frac{1}{2}, \\ (E_2(x) &= x^2 - x)^2, \\ E_3(x) &= x^3 - \frac{3}{2}x^2 + \frac{1}{4}. \end{aligned}$$


```

The use of `\sfr` had no effect!

4.19.3 Either way.

TB196 gives an example for flushing both ways. Better suited for 2-column format is

$$\sin z = z - \frac{z^3}{3!} + \frac{z^5}{5!} - \frac{z^7}{7!} + \frac{z^9}{9!} - \frac{z^{11}}{11!} + \dots$$

$|z| < \infty.$

obtained via

²³In the math book there was no squaring, the unequal line distances were visible without it. I could not reproduce that.
²⁴Look also at the prime-ry section.

```


$$\sin z = z - \frac{z^3}{3!} + \frac{z^5}{5!} - \frac{z^7}{7!} + \frac{z^9}{9!} - \frac{z^{11}}{11!} + \dots$$


```

Note the use of the empty formula after `\hfill`, in order to make T_EX recognize an eventual '+' symbol as binary operator, and provide suitable spacing.

4.19.4 Centered and right.

TB ex19.17, gives a set of formulae with one formula splitted into two parts, and the second part appropriately placed relative to the first part. The solution has made use of `eqalign`. In order to disturb as little as possible the appearance of the aligned set of equations one could wish to flush right the splitted parts. This is in agreement with Swanson(1986, 3.3.5). In 2.5.2 she advises to slash stacked fractions in display because of space economy. The example was adapted from the math book. From the context it is clear why this representation had been chosen in the math book.

Example (*Splitted parts flushed right*)

$$\begin{aligned} u_k &= \frac{1}{(k+1)(2k+1)(4k+1)} \\ &= \frac{\frac{1}{3}}{k+1} - \frac{1}{k+\frac{1}{2}} + \frac{\frac{2}{3}}{k+\frac{1}{4}} \\ &= \frac{1}{3} \left(\frac{1}{k+1} - \frac{1}{k} \right) - \left(\frac{1}{k+\frac{1}{2}} - \frac{1}{k} \right) \\ &\quad + \frac{2}{3} \left(\frac{1}{k+\frac{1}{4}} - \frac{1}{k} \right) \quad (3) \end{aligned}$$

$$\sum_{k=1}^{\infty} u_k = -\frac{1}{3}\psi(2) + \psi(1\frac{1}{2}) - \frac{2}{3}\psi(1\frac{1}{4})$$

is obtained by

```


$$\sum_{k=1}^{\infty} u_k = -\frac{1}{3}\psi(2) + \psi(1\frac{1}{2}) - \frac{2}{3}\psi(1\frac{1}{4})$$


```

4.20 Interrupts.

Authors want to connect displayed, and aligned, formulae by texts. T_EXnically this means the opposite: the alignment has to be interrupted. For example, TB193,

$$x = y + z$$

and

$$x^2 = y^2 + z^2$$

is obtained by

```

\eqalignno{x&=y+z\cr
\noalign{\noindent{\rm and}}
x^2&=y^2+z^2\cr}

```

Note the difference when `\eqalign` is used, TB ex19.14, and don't forget `\noindent`.²⁵

I was trapped when I footnoted the text 'and'. The mark appeared on its own.

4.21 On your own.

Yes, you can be in complete control and escape the automatism. From the math book I got the impression that `\eqalign` was used throughout, and all formulae were centered.

'For other displays, plain T_EX provides `\displaylines`, which lets you display any number of formulae in any way you want, without any alignment.'

Incidental left justification can be obtained easily, TB194, ex19.16. The T_EXfall is to adhere to `\displaylines` throughout a T_EXscript, exercising typographic mark-up, at the price of cumbersome adaptation, and unnecessary exposure to the danger of being inconsistent.

Example (*Left now and then*)

$$P_\nu(\cos \theta) \quad (\text{MD})$$

is obtained via

```

\displaylines{\indent P_\nu(\cos\theta)
\hfill\llap{(MD)}\cr}

```

Note however that in the MAPS of NTG `\parindent=0`.

4.22 Generality.

From TB375 we have

'The goal is to set T_EX up so that the respective constructions `$$\alpha$$`, `$$\alpha\leq\beta$$`, and `$$\alpha\leq\beta\leq\alpha$$` will cause a macro `\generaldisplay` (in `display`) to be invoked, with `\eq` to be defined α ; furthermore the test `\ifeqno` should be true when an equation number β

is present, and `\ifleqno` should be true in the case of `\leqno`. When β is present, it should be stored in `\eqn`. Here α and β are arbitrary balanced token lists that don't contain either `\eqno` or `\leqno`.'

4.22.1 Lefties forever.

The pitfall for using left justification throughout the publication is that with short formulae the resulting pages look 'too white' at the right half.

Needed are the macros provided at TB376 augmented with a suitable `\generaldisplay` macro for left alignment, indented with `\parindent`, and formulae numbers flushed right. No adaptation of the T_EXscript!²⁶

$$f(x) = \begin{cases} 0, & \text{if } x < 0 \\ 1, & \text{if } x \geq 0 \end{cases} \quad (\text{Stepfunction})$$

is obtained via, TB376

```

\def\cases{0,&\if\quad $x<0$\cr
1,&\if\quad $x\geq0$\cr}
\eqno{\rm Step function}
%with declarations
\newif\ifeqno \newif\ifleqno
\everydisplay{\displaysetup}
\def\displaysetup#1$$\{
\displaytest#1\eqno\eqno\displaytest}
\def\displaytest%
#1\eqno#2\eqno#3\displaytest{%
\if!#3!
\ldisplaytest#1\leqno\leqno\ldisplaytest
\else\eqnotrue\leqnofalse
\def\eqn{#2}\def\eq{#1}%
\fi
\generaldisplay}
\def\ldisplaytest%
#1\leqno#2\leqno#3\ldisplaytest{%
\def\eq{#1}
\if!#3!\eqnofalse
\else\eqnotrue\leqnotrue\def\eqn{#2}\fi}
\def\generaldisplay{%
\netdpw\hsize
\advance\netdpw-\parindent%Compenstae
\leftline{\indent$\displaystyle\eq$
%No hfill, to allow eqn no in eqalign
%to be right adjusted
\ifeqno\hfill\llap{$\eqn$}\fi}
%Termination display
}%end generaldisplay

```

Note the use of the parameter separators, called sentinels in traditional programming.

Partial solution.

For the case the T_EXscript does not contain `\eqno` tags a simpler solution is provided in TB ex19.4. The solution given does not allow for `\eqalignno` to be used either.

²⁵ Again a source of confusion.

²⁶ `\eqalignno` can't be used, however, yielding an error message. `\eqalignno` can be adapted by: use `\vcenter`, give the first `\tabskip` the value zero, and provide the `\halign` with the size `\displaywidth-\parindent`.

4.23 Graphs.

The \TeX fall is that \TeX is weak with respect to graphics. However, via the insert mechanism space can be left open to paste in figures and the like, made separately and independently by other tools. Electronic paste-up at the Postscript level is possible. To let text flow around a figure (open space) can be done. See Cork90 proceedings for details.

One easily dives into picture like environments, and then the too many pitfall may open up. In TB ex18.46 commutative diagrams are dealt with. Simple diagrams can be built from there. As example we take the diagram showing the calculation of the autocorrelation, either via $a_f = f \otimes f$, or by means of Fourier transform, followed by multiplication, and the inverse transform. The following is derived from TB ex18.46, 358, where \mathcal{F} denotes Fourier transform and \mathcal{F}^{-1} the inverse Fourier transform.

$$\begin{array}{ccc} f & \xrightarrow{\otimes} & a_f \\ \downarrow \mathcal{F} & & \uparrow \mathcal{F}^{-1} \\ F(f) & \xrightarrow{\times} & (\mathcal{F}(f))^2 \end{array}$$

obtained via

```
\def\llongrightarrow{\relbar\joinrel
\relbar\joinrel\rightarrow}
\def\normalbaselines{
\baselineskip20pt
\lineskip3pt
\lineskiplimit3pt}
\def\mapright#1{\smash{\mathop{
\llongrightarrow}\limits^{#1}}}
\def\mapdown#1{\Big\downarrow
\rlap{\$vcenter{\hbox{\$#1\$}}\$}}
\def\mapup#1{\Big\uparrow
\rlap{\$vcenter{\hbox{\$#1\$}}\$}}
\matrix{f&\mapright\otimes&a_f\cr
\mapdown{\cal F}&&\mapup{\%
\cal F}\strut^{-}\cr
F(f)&\mapright\times&\bigl(
\cal F)(f)\bigl)^2\cr}
\def\normalbaselines{
\baselineskip20pt
\lineskip3pt
\lineskiplimit3pt}
```

Borceux(1990) has pointed out that diagonal as well as unequal length connectors are needed. The latter because of the unequal size of the ‘knots’. The interested reader is referred to Borceux(1990). For curved connectors the Bezier technique might be used. $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\TeX$ is rich with respect to commutative diagrams, among others.

4.24 All in the family.

A fonts \TeX fall is related to using a $\hbox{\dots}$. To quote from TB163

‘But such uses of $\hbox{\dots}$ have two disadvantages: (1) The contents of the box will be typeset in the same size, whether or not the box occurs as a subscript; for

example, ‘ $\$x_{\hbox{\max}}\$$ ’ yields x_{\max} . (2) The font that’s used inside $\hbox{\dots}$ will be the ‘current font,’ so it might not be roman.’

In the \TeX script I found $\def\Re{\hbox{\Re}}$, adhering to disadvantage (2). See also TB ex19.1 for the unexpected italic result.

Plain provides basically (computer codern) roman, bold, slanted, text italic, typewriter type, math symbols and math extensions. Most fonts are available in 10pt, 7pt and 5pt. Especially the math extension is of interest, because of the composition possibilities. For example the ‘n-sized’ open parenthesis is composed of the entries ‘60 + (’102)ⁿ + ’100. Because of mnemotechnique and because of the composition process, the symbols have been made available via names. The ordering is prompted by the functionality into, TB434 etc.

1. Lower case Greek letters,
 2. Upper case greek letters,
 3. Calligraphic capitals,
- etc.

For example $f \mapsto g$ is obtained via $\$f\mapsto g\$$. BLUE does not have to worry about font tables, not even for symbols of varying size. Some understanding does not harm however. Curious are the names for \wedge , and \vee . No \and , respectively \or . This is understandable because \or is already in use in the case construct. The latter can be part of an expression, and therefore overloading is not possible for \or .

Now and then other symbols than those provided in the font tables of Appendix F are wanted. Occasionally these can be constructed, like the symbols denoting the sets of natural numbers, \mathbb{N} , integers, \mathbb{Z} , rational numbers, \mathbb{Q} , reel numbers, \mathbb{R} , and complex numbers, \mathbb{C} .

These are obtained by kerning — ‘poor man’s blackboard bold.’ My version is

```
\def\N{\rm I\kern-.5ex N}
\def\Z{\rm Z\kern-.9ex Z}
\def\Q{\rm\kern.2ex\vrule
height1.3ex depth-.1ex
width.4pt\kern-.7ex Q}
\def\R{\rm I\kern-.5ex R}
\def\C{\rm\kern.3ex\vrule
height1.3ex depth-.1ex
width.4pt\kern-.7ex C}
```

Generally they are already available somewhere, see Quin(1990), and especially AMS(1990). The math book has used it its own brand of poor man’s blackboard bold. Swanson(1986, 2.4.8e) advises just to use boldface.

I also needed \gtrless .

```
\def\gtrless{\mathrel{\vcenter{
\hbox{\$buildrel\textstyle{>}
\over{>}\$}}}}
```

Note the use of \textstyle , otherwise the cramped \textstyle is obtained.

In \TeX HaX90.20, Duchier(1990) has published a macro for the external tensor product, \boxtimes .

A simplified version reads

```
\def\boxit#1{\vbox{\hrule\hbox{% Necessary
\vrule#1\vrule}\hrule}}
\def\boxtimes{\mathbin{\boxit{\$ \times \$}}}
```

Fortunately, these kinds of symbols are now available in AMSfonts, for free.

In general one needs to know what is locally available and how it looks like. For the latter there exist a program called ‘testfont,’ which prints the font table with the symbols in it. Once the fonts have been selected they must be made known to T_EX. Next one can integrate the fonts into families, such that automatically in displaystyle, textstyle, scriptstyle and scriptscriptstyle the right size appears. More perfect is integration into size-switching macros analogous to \tenpoint, \ninepoint, respectively \eightpoint, for the T_EXbook itself, see Appendix E. The size-switching macros for the TB contain the families: \itfam, \slfam, \ttfam, and \bffam. How to do this is given on TB414.

Beyond the scope of this paper is construction of symbols via METAFONT, as well as the system managers work to get fonts from elsewhere and install these.

5 The good news

In order to go forward classical items like: Better user interfaces, Education, Have it done, and Support, are needed.

Because of T_EX, and e-mail, I could work in the spirit of Swanson(1986)

‘Perhaps some day a typesetting language will become standardized to the point where papers can be submitted to the [publisher] from computer to computer via telephone lines. Galley proofs will not be necessary, but referees and/or copy editors could send suggested changes to the author, and he could insert these into the manuscript, again via telephone.’

5.1 Better

In contrast with public domain (plain) T_EX the better user interfaces do cost. Appealing names are in use: The writer’s workbench, Publishing Environment, and the new vogue DTP. Generally, they support inputting the (math)script and providing laser printer output. The user does not have to know that T_EX is used as formatter, even worse, T_EX is occasionally lacking. When T_EX is used the T_EX file can be accessed, generally.

The user interfaces I have seen, are weak when corrections have to be made. I also noticed limited context sensitivity. As example of the latter I asked a vendor to use a matrix as an integrand. The integral sign did not grow with the size of the matrix. I don’t expect those systems to allow for explicit formatting commands. With

nested parentheses, fences and the like, different sizes have to be ordered for explicitly, now and then. For example

$$\|a(x + y)\| \leq |a| \|x + y\| \leq |a| (\|x\| + \|y\|),$$

is a typist pitfall as well as a pitfall for automated SGML-based user interfaces, I presume.

5.2 A world of learning.

T_EX can be learned from the T_EXbook with a T_EXnigma at hand. It is easier and more economical to be guided by an experienced teacher. TUG traditionally, and recently many LUGs, organize a variety of T_EX, L^AT_EX, and Metafont related courses. Although discussion has started about what the various courses should provide and how they should be related; no agreement has emerged of yet, see Childs(1989a,b), and van der Laan(1989). I agree with Martin(1990) that a class consisting of T_EXnical typists needs another approach than a class of scientists. Everybody needs T_EX etc. intelligent editors. I have heard of L^AT_EX-taylored EMACS and of enhanced EDT, Williams and Hall(1990). Education is paramount. Why not release THE video tapes at cost? Imagine, PD T_EX on your PC, the teaching on video, all that at home!

It might be clear that I still gaze at the quotations on TB159, . . . somewhat in unbelieve.

‘The learning time is short. A few minutes gives the general flavor, and typing a page or two of a paper generally uncovers most of the misconceptions. —Kernighan and Cherry, A system for Typesetting Mathematics (1975)’

‘Within a few hours (a few days at most) a typist with no math or typesetting experience can be taught to input even the most complex equations. —Peter J. Boehm, Software and hardware considerations for a technical typesetting system (1976)’

5.3 Service.

A professional typist is better suited for typing mathscripts than an author, despite not being a mathematician and not understanding the contents. AMS provides T_EX typing services, see AMS(1990). I conjecture that more such services are needed, at reasonable price.

The demand is not (yet) large, I guess, because authors consider it a challenge to T_EX their documents themselves, at the expense of ample trial-and-error. Besides, the author understands what is going on, likes to remain in complete control, especially when the proofing is cumbersome because of the typist suffering from insufficient T_EXknowledge.

On the other hand, most documents enjoy a local readership and the obtained (form) quality in print —via naive T_EXing or via another non-optimal tool—is considered

sufficient in relation to the contents, the document preparation know-how of the typist, the readership, and the life-time of the document.

When publishing an article or book is in sight, a publishing house might take over and provide professional typing service, if needed, apart from other quality warranting issues.

5.4 Lean upon.

Sooner or later typesetting challenges will be encountered. Of course one can puzzle and find out eventually oneself, but it is generally more economical to consult a guru, despite the salaries. TUGboat, and electronic digests, among others, pay attention to queries from their readers. Listservers (and the digests) opened the possibility to query a community instead of one person. Besides, one can always have the problem ‘turnkey’-solved by hiring a programmer-consultant.

Jam session

\TeX ing a mathscript, lacking format commands, is too difficult for a non- \TeX -trained typist. Moreover the typing task is silently augmented because proofs are more difficult to provide. It is unclear in what way AMS \TeX typing services fills up the gap. It is not true that once one can talk math by phone, one can \TeX math. One must not only write e.g. ‘:’, but also specify whether it is used as punctuation symbol or as an operator. A mathematician must be told to specify these kinds of things in his mathscript, providing guidance for the \TeX ist.

On the other hand authors must not persuade \TeX ists into the typography mark-up \TeX fall by supplying underlining, bold etc. wishes.

For \TeX ing math, both the author and the \TeX ist need to be aware of the possibilities of \TeX , and the consequences for the mathscript.

\TeX ophil advisors are considered harmful, demonstrating ‘a little knowledge is dangerous.’ The more so because only the best is good enough.

In the hands of mathematicians, \TeX etc. is challenging and enslaving. From the math book I understand that using \TeX costs already so much energy, that it is difficult to adhere to consistency. Early \TeX scripts suffer from various \TeX falls. Nevertheless, the result in print is considered good enough, because of lack of better small-scale alternatives.

\TeX is a wonderful, but unusual tool. It challenges. I admire the design. I never read a manual so many times. I love \TeX ! Grace to the indefinite lifetime, investing in learning \TeX is worthwhile.

But, . . . \TeX is non-robust and error-prone. Beware! Because of the complexity, the freedom, and flexibi-

lity, augmented with unawareness of typographic tradition, one can easily err—and, whether one likes it or not—err, and err again, $\text{D}\text{E}\text{K}(1989)$.²⁷ And if ever

The quality of the results depends on what you,
yourself,
make out of it.

The difficulty with rule-books is that they lag behind new \TeX nology.

References

- [1] AMS A manual for authors of mathematical papers (1973): Booklet, 24p.
- [2] AMS \TeX Publishing Services (1990): Leaflet.
- [3] $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\TeX$ (1990, 2.0): User’s Guide.
- [4] $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\TeX$ (1990, 2.0): Installation Guide.
- [5] AMSFonts (1990, 2.0): User’s Guide.
- [6] AMSFonts (1990, 2.0): Installation Guide.
- [7] Beeton, B.N. (1990): TUGboat production: \TeX , \LaTeX and paste-up. Appendix S. Minutes 5th NTG meeting. (Talk presented at SGML- \TeX conference Groningen, 1990.)
- [8] Borceux, F. (1990): De la construction de diagrammes. Cahiers GUTenberg, 5, 41–48.
- [9] Cameron, P.J. (1990): Adventures of a new \TeX nician. \TeX line 12, 2–3.
- [10] Cheswick, B. (1990): A permuted index for \TeX and \LaTeX . CST145, AT&T Bell Laboratories.
- [11] Childs, B. (1989): Teaching \TeX . TUGboat 10, 2, 156–163.
- [12] Childs, B. (1989): Answers to \TeX tests. TUGboat 10, 3, 319–323.
- [13] Duchier, D. (1990): \TeX , external tensor product. \TeX HaX90.20.
- [14] Haeringen, H. van (1990): Priv. comm.
- [15] Knuth, D.E. (1984): The \TeX book. Addison-Wesley. (\TeX , frozen in version π , 3.1415. . . is treated in Vol A of Computers and Typesetting from the *ninth* printing onwards, and in \TeX book from *seventeenth* printing onwards. (And if you can’t remember the numbers, look for the `\language` command in the index.)
- [16] Knuth, D.E. (1985): Recipes and fractions. TUGboat 6, 1, 36–38.
- [17] Knuth, D.E. (1989): The errors of \TeX . Softw-Prac&Exp. 19, 7, 607–685.

²⁷I wonder, if error-free \TeX ing can ever be attained. Once in a BLUE moon?

- [18] Knuth, D.E., T. Larrabee, P.M. Roberts (1989): Mathematical writing. MAA Notes 14. Stanford University (STAN-CS-88-1193). (With contributions by P. Halmos, Leslie Lamport, Mary-Claire van Leunen, Nils Nilsson, Rosalie Stemer, Jeff Ullman, and Herbert S. Wilf.)
- [19] Lamport, L. (1986): \LaTeX , a document preparation system. User's guide & reference manual. Addison-Wesley.
- [20] Martin, C.R. (1990): \TeX for \TeX nical typists. TUGboat, 11, 3, 425–428.
- [21] Laan, C.G. van der (1989): Teaching \TeX : critics and \LaTeX proposal. Appendix V. Minutes 4th NTG meeting.
- [22] Nearing, J. (1989): Extended equation numbering in plain \TeX . TUGboat, 10, 1, 82–88.
- [23] Pittman, J.E. (1988): Equation numbering in plain \TeX . TUGboat, 9, 3, 287–288.
- [24] Quin, L.R.E. (1990): Summary of Metafont fonts available. \TeX MaG, 4, 6.
- [25] Spivak, M.D. (1986): The Joy of \TeX . A gourmet guide to typesetting with the $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX macro package. Addison-Wesley.
- [26] Swanson, E. (1986): Mathematics into type. AMS. (Copy editing and proofreading of mathematics for editorial assistants and authors.)
- [27] Temme, N.M. (1990): Speciale functies in de mathematische fysica. Epsilon 15. ISBN 90-5041-019-7. (Dutch)
- [28] Whitney, R.F, B.N. Beeton (1989): TUGboat author's guide. TUGboat, 10, 3, 378–385.
- [29] Wichura, M.J. (1990): Showing-off math macros. TUGboat 11, 1, 57–61.
- [30] Williams, L, L. Hall (1990): Increased efficiency using advanced EDT editing features. TUGboat, 11, 3, 421–424.

Appendix

Contents Math into BLUES

Prelude
1 Traditional math page make-up
2 Am I blue?
3 What is wrong Doc?
4 The bad news
4.1 Lessened SoC
4.2 Too many
4.3 First gains
4.4 Emptyness
4.5 Class unawareness
4.5.1 Innocent braces
4.5.2 Whoops
4.5.3 Just a comma
4.5.4 With dots
4.6 CMR
4.7 Kameleons
4.8 Triads
4.8.1 Real life
4.9 Accents differ
4.9.1 Embellishments
4.9.2 Prime-ry
4.10 $\frac{1}{2}$'s never have the right size
4.11 Various OOOOO's
4.12 Backslash penances
4.13 Over and over
4.14 Too difficult
4.15 As simple as math can be
4.15.1 Automatisms
4.16 You name it
4.17 Both ways
4.18 We want more
4.18.1 Adaptation of <code>\eqalign</code>
4.18.2 Adaptation of <code>\(1)eqalignno</code>
4.19 Loooooooooooooooooooo
4.19.1 Via consecutive displays
4.19.2 Via eqalign
4.19.3 Either way
4.19.4 Centered and right
4.20 Interrupts
4.21 On your own
4.21.1 Generality
4.21.2 Lefties forever
4.22 Graphs
4.23 All in the family
5 The good news
5.1 Better . . .
5.2 A world of learning
5.3 Service
5.4 Lean upon
Jam session
References

BIJLAGE S

Babel, a multilingual style-option system for use with \LaTeX 's standard document styles^{1 2}

Johannes Braams

PTT Research Neher Laboratories
 P.O. Box 421
 2260 AK Leidschendam
 jl.braams@pttrnl.nl

March, 1991

Abstract

The standard distribution of \LaTeX contains a number of document styles that are meant to be used, but also serve as examples for other users to create their own document styles. These styles have become very popular among \LaTeX users. But it should be kept in mind that they were designed for American tastes and contain a number of hard-wired texts. This article describes a set of document-style options that can be used in combination with the standard styles, which makes the latter adaptable to other languages.

1 Introduction

Although Leslie Lamport has stated [5] that one should not try and write *one* document-style option to be used with *all* the standard document styles of \LaTeX , that is exactly what I have done with this system of style options. The reasons for this approach will be explained in section 2.

A lot of the ideas incorporated in this set of files come from the work of Hubert Partl [4], `german.tex`. Some parts in the implementation are different, others are the same. It will be shown that `german.tex` can be modified to fit into this scheme of style options.

2 Why Babel?

When I first started using \LaTeX I was very happy with just the style files that are distributed with the standard distributions of \TeX and \LaTeX . That means, as long as I made texts in English I was happy. Then as other users found out about \LaTeX and its advantages, they started using it for texts in other languages than English. As I was the most experienced \LaTeX user at the time, they came to me and asked me ‘When I’m writing a report in Dutch I don’t want chapters to be named “Chapter”, I want them to be named “Hoofdstuk”, how do you change that?’. At that time I didn’t know, but

I soon found out. The first thing I found was that Leslie Lamport states in [2, pages 85–86] that you have to redefine the command `\@chapapp` to get the desired result. This looked rather promising to me, so I had a look at the style files to find out how other such strings as “Figure” might be redefined. It was then that I found out that `\@chapapp` is the *only* string defined this way, whereas all others are hard-wired into the style.

My first solution to this problem was to create a new document style file called `artikel.sty` as a “Dutch” counterpart to `article.sty`. The same was done for `report.sty`. This is exactly what Leslie Lamport suggests in [5]. This approach has one major drawback however: you get two copies of basically the same file to maintain. This was discovered when newer releases of the styles reached our site. The standard styles had to be replaced *and* edited all over again to get the “Dutch” versions back. About the same time, in early 1988, a discussion on this subject appeared in \TeX hax. One of the persons commenting was Hubert Partl. The method he suggested was to modify the standard document styles by replacing the hard-wired texts by macros such as `\@chapapp`. This led me to my second attempt at a solution. I modified the standard styles (all four of them) as suggested, but while doing that added an option, implemented like the option `draft`, by defining a command `\ds@dutch`. This command would

¹During the development ideas from Nico Poppelier and Piet van Oostrum have been used.

²To be submitted to TUGboat; © 1991, \TeX Users Group.

set a variable to indicate which language was requested. This variable I used later on in a `\case` statement. In this `\case` statement a choice is made between English, Dutch and possibly other languages for texts such as “Figure” and “Contents”. Unfortunately, some of this implied changing the secondary style files `xxx10.sty`, `xxx11.sty` and `xxx12.sty`. This was unfortunate because one of the research groups in our laboratories complained their document style didn’t work properly. It turned out that their style was a modified `article.sty` that had been given a different name, but it still loaded `art10.sty` etc. I found a temporary solution, but I still wasn’t exactly happy with the situation. Besides this, the drawback of replacing the document styles with newer versions still existed.

When after a while a new version of the \LaTeX distribution arrived at our site, I began to think about a different way to solve the problem. In the meantime Hubert Partl had his `german.sty` published in *TUGboat* [4]. His article pointed the way to a different solution. Triggered by the discussion in `TeXhax` in early 1989 about how to detect which is the main (primary) style when processing a document, I started work on what is now available as `dutch.sty` version 1.0, dated may 1989³. While working on this style option I discovered that some parts could be borrowed from `german.sty`. This ‘discovery’ and some discussions I had with others at EuroTeX89, the fourth European \TeX Conference, held in september 1989 in Karlsruhe, led me towards a more universal approach. The basic idea behind it was, starting from the algorithm to detect the main style, to design an approach with one common file that contained macro definitions needed by a number of language-specific style options. Users specify the name of any of these language-specific options as an option to the `\documentstyle` command, and internally the common file is read.

3 \LaTeX and document-style files

Before I discuss some of the code in the `babel` system I would like to discuss the document-style mechanism used by \LaTeX . Every \LaTeX document should start with a line like:

```
\documentstyle[opt1,opt2,...]{docstyle}
```

This line of code instructs \LaTeX to first load the file `docstyle.sty`. When that is done the ‘options’ are processed *in the order specified*, by reading the files `opt1.sty`, `opt2.sty`, etc. This implies that definitions, made in the file `docstyle.sty` can be overridden in one of the option files. It is even possible to redefine code from the very kernel of \LaTeX , but you have to know what you are doing.

Some care has to be taken in writing document-style options, because a number of problems can occur. First of all, if a document-style option should be modest in size,

if it tries to redefine most of the code in `docstyle.sty` I think you should write (and maintain) your own, complete, document style. Next, as it was possible to override definitions from the main file in an option file, it is of course also possible to override definitions made in another option file. When this happens, your document might depend on the order in which you have specified your document-style options.

This mechanism of overriding definitions from the main document style is exploited in the `babel` system. The macros that contain the hard-wired texts are redefined in the common part of `babel`, replacing each of these texts with a unique macro. These macros have to be defined in the language-specific files.

4 \LaTeX and multilingual documents

In a european environment it sometimes happens that one wants to write a document that contains more than *one* language. I have an example of a document, published by the EEC, that contains 9 (nine) different languages. Also in linguistics one can find documents written in more than one language, i.e. to compare two languages.

If you have to write such a multilingual document you should try to conform to the typographical conventions in use for each language. A well known example is the type of quotation marks used. \TeX supplies the user with “quoted text”, but a Dutch user might want to have „quoted text”, whereas a German text should contain „quoted text“ and a frenchman would perhaps like to see something like «quoted text». These language specific conventions should be implemented in a document-style option file for each language. These files should then be useable with *all* document styles.

In such a multilingual document a user would specify the languages used as options to the `\documentstyle` command. He would also want a mechanism to be able to switch between these languages in a simple way. When he would use \TeX version 3.0 for the processing of his document, he would also want the hyphenation to come out right for the different languages.

5 Overview of the babel solution

5.1 The core of the system

The problems described in sections 3 and 4 can be solved using the `babel` system of document-style options.

The core of this system currently performs three functions.

1. It defines a user interface for switching between languages;

³This file is available from `listserv@hearn` as file `dutch.old`.

2. It contains code to dynamically load several sets of hyphenation patterns;
3. It ‘repairs’ the document styles provided in the standard distribution of \LaTeX .

Obviously part 2 can only be used while running iniTeX to create a new format, whereas part 3 should *not* be read by iniTeX . Part 3 should even disappear when \LaTeX version 3.0 arrives, as the style files supplied with the new \LaTeX will no longer be language specific. Part 1 can either be loaded into the format with multiple hyphenation patterns, or it can be read while processing a document.

For this reason the core of the `babel` system is stored in two separate files, `babel.switch`, containing parts 1 and 2, and `babel.sty` which contains part 3. The file `babel.sty` will instruct \LaTeX to load `babel.switch` if necessary, the file `babel.switch` checks the format to see if hyphenation patterns *can* be loaded.

5.2 Language specifics

The language switching mechanism contains a couple of hooks for the developers of language-specific document-style options.

First of all the macro `\originalTeX` should be defined. Its function is to disable special definitions made for a language to bring \TeX into a ‘defined’ state. A language-specific document-style option might, for example, introduce an extra active character. It would then also modify the definitions of `\dospecials` and `@sanitize`. Such an option would then define a macro to restore the original definitions of these macros and restore the extra active character to its normal category code. It would then `\let \originalTeX` to this ‘restoration’ macro.

To enable the language-specific definitions three macros are provided in the switching mechanism, `\captions⟨language⟩`, `\date⟨language⟩` and `\extras⟨language⟩`.

The macro `\captions⟨language⟩` should provide definitions for the macros that replaced the hard-wired texts in the document style and the macro `\date⟨language⟩` should provide a definition for `\today`. The real fun starts with the macro `\extras⟨language⟩`. This macro should activate all definitions needed for `⟨language⟩`.

6 The user interface

The user interface to the `babel` system is quite simple. He should specify the languages he wants to use in his document in the list of document-style options. For instance, for a document in which both the English and the Dutch language are used, the first line could read:

```
\documentstyle[a4,dutch,english]{artikel1}
```

Please note that in this case the Dutch-specific definitions are inactive when \LaTeX has finished processing document-style option files.

If the user then wants to switch from English to Dutch he would include the command

```
\selectlanguage{dutch}
```

before starting to write Dutch.

If a user wants to write a document-style option of his own he might like to define a macro that checks which language is in use at the time the macro is executed. For this purpose the macro `\iflanguage⟨language⟩⟨then-clause⟩⟨else-clause⟩` is available.

7 Implementation of the core of the system

In this section I would like to discuss some parts of the implementation of the `babel` system. Not all code will be shown, because some parts of it are just series of slightly modified code from the standard document styles. The files are fully documented and interested readers can print them if they have access to the `doc` option, described by Frank Mittelbach.

The description of the macros that follows is based on an environment using \TeX 3.x, together with a version of `lplain.tex` based on `plain.tex` version 3.x. The actual implementation allows for other situations as well, i.e. a version of `babel.sty` for \TeX 2.x will be available.

7.1 Switching languages

For each language to be used in a document a control sequence of the form `\l@⟨language⟩` has to be defined. This will either be done while loading hyphenation patterns or while loading the language-specific file. The implementation of `\selectlanguage{⟨language⟩}` and `\iflanguage{⟨language⟩}{ }{ }` is based on the existence of `\l@⟨language⟩`

To switch from one language to another the macro `\selectlanguage` is available. Its definition can be seen in figure 1. The first action it takes is to check whether the `⟨language⟩` is known, if it is not an error is signalled. If the language is known `\originalTeX` is called upon to reset any previously set language-specific definitions. Next the register `\language` is updated and the three macros that should activate all language-specific definitions are executed. Finally the macro `\originalTeX` receives a new replacement text in order to be able to deactivate the definitions just activated.

The macro `\iflanguage` (see figure 2) will issue a warning when its argument is an ‘unknown’ language. It

```

\def\selectlanguage#1{%
  \@ifundefined{l@#1}
    {\@nolanerr{#1}}
    {\originalTeX
     \language=\expandafter\csname l@#1\endcsname\relax
     \expandafter\csname captions#1\endcsname
     \expandafter\csname date#1\endcsname
     \expandafter\csname extras#1\endcsname
     \gdef\originalTeX{\expandafter\csname noextras#1\endcsname}
    }
}

```

Figure 1: The definition of `\selectlanguage`.

```

\def\iflanguage#1#2#3{%
  \@ifundefined{l@#1}
    {\@nolanerr{#1}}
    {\ifnum\language=\expandafter\csname l@#1\endcsname\relax
     #2 \else #3
    \fi}
}

```

Figure 2: The definition of `\iflanguage`

then goes on to compare the value of `\language` and `\l@{language}`.

7.2 Dynamically loading patterns

With the advent of \TeX 3.0 it has become possible to build a format with more than one hyphenation pattern preloaded. The core of the babel system provides code, to be executed by *ini \TeX* , to dynamically load hyphenation patterns. The only restriction is that the implementation of \TeX that you use has to have rather high settings of `trie_size` and `trie_op_size` to actually load several hyphenation patterns.

For the purpose of dynamically loading hyphenation patterns a ‘configuration file’ has to be introduced. This file will be read by *ini \TeX* . Each line should contain either a comment, nothing or the name of a language and the name of the file that contains the hyphenation patterns for that language. In figure 3 an example of such a file, instructing *ini \TeX* to load patterns for three languages, English, Dutch and German.

The configuration file will be read line by line using \TeX ’s `\read` primitive. Because the name of a file might be followed by a space-token and comment (as in the example) a macro to process each line is needed. The definition of this macro, `\process@language`, can be found in figure 4.

The macro should strip all spaces following its arguments. Its first argument is used to define `\l@{language}`. The macro `\addlanguage` is basically a non-outer version of the plain \TeX macro `\newlanguage`. The second argument of

`\process@language` is the name of the file containing the hyphenation patterns. Before the file can be read, the register `\language` has to be updated.

The configuration file is read in a `\loop` (see figure 5). When a record is read from the input file a check is done whether the record was empty. If it was not, a space token is added to the end of the string of tokens read. The reason for this is that we have to be sure there always is at least *one* space token present. (A space token is stored in `\toks0`.) When that has been taken care of the data just read can be processed. The last thing to do is to check the status of the input file, in order to decide whether \TeX has to continue processing the `\loop`. When all patterns have been processed the value of `\language` is restored.

7.3 ‘Repairing’ \LaTeX ’s standard document styles

A large part of the core of the babel system is dedicated to ‘repair’ the standard document styles. This means redefining the macros in table 1.

As an example of the way the macros have to be redefined, the redefinition of `\tableofcontents` is shown in figure 6.

The standard styles can be distinguished by checking the existence of the macros `\chapter` (not in *article* and *letter*) and `\opening` (only in *letter*). The result of these checks is stored in the macro `\doc@style`. When `\doc@style` already exists (which is the case when for instance *artikel1.sty* is used [7]) it is not superseded (see figure 7).

```
% File      : language.dat
% Purpose   : tell iniTeX what files with patterns to load.
english    english.hyphenations

dutch      hyphen.dutch % Nederlands
german     hyphen.ger
```

Figure 3: An example configuration file

```
\def\process@language#1 #2 {%
  \expandafter\addlanguage\csname l@#1\endcsname
  \expandafter\language\csname l@#1\endcsname
  \input #2}
```

Figure 4: The definition of `\process@language`.

8 Implementing a language specific document-style option file

To illustrate the way a language specific file can be implemented the file `dutch.sty` is discussed here. Note that not all of the code contained in the file `dutch.sty` is shown here, only those parts that are of interest for the scope of this article are included. If the reader would like to see the complete code, he can print all files in the `babel` system, using the file `doc.sty`, described by Frank Mittelbach in [6].

8.1 Compatibility with plain T_EX

The file `german.tex` [4] was written in such a way that it can be used by both plain T_EX users and L^AT_EX users. This seemed a good idea, so all files in the `babel` system can be processed by both plain T_EX and L^AT_EX. But some of the “useful hacks” from L^AT_EX are used, so for a plain T_EX user they have to be defined. For this purpose the format is checked at the start of a language specific file. If the format is `plain` an extra file, called `latexhax.tex` is read.

This file should be read only once, so another check is done on the existence of one of the commands defined there.

A new group is started to keep the definition of the macro `\format`, which is used in the following if statement, local. When the current format turns out to be plain T_EX

the file `latexhax.sty` has to be read. But the definitions in that file should remain valid after the group is closed. This could be accomplished by making all definitions `global`, but another solution is to tell T_EX to process the file `latexhax.sty` *after* the current group has been closed. The command `\aftergroup` puts the next token on a list to be processed after the group.

8.2 Switching to the Dutch language

In section 7.1 the names of macros needed to switch to a language have been described. In figure 9 these macros and their definition are shown for the Dutch language.

The definitions of `\captionsdutch` and `\datedutch` are pretty straightforward and need not be discussed. The macro `\extrasdutch` will be discussed in some more detail.

First, because for Dutch (as well as for German) the " character is made active, the L^AT_EX macros `\dospecials` and `\@sanitize` have to be redefined to include this character as well. The new definitions are implemented as two special commands, so we globally `\let` the originals to their new versions. Then the " character is made active and is defined. Then, to prevent an error when `\` appears in a moving argument, the macro `\` is redefined and made robust. All this is done inside a group to keep the category code change for the " character local.

```
\loop
  \readl to \@config@line
  \ifx\@config@line\empty
  \else
    \edef\@config@line{\@config@line\the\toks0}
    \expandafter\process@language\@config@line
  \fi
  \ifeof1 \@morefalse \fi
  \if@more\repeat
\language=0
```

Figure 5: Reading the configuration file line by line

macro	article	report	book	letter
<code>\fnum@figure</code>	x	x	x	x
<code>\fnum@table</code>	x	x	x	x
<code>\tableofcontents</code>	x	x	x	
<code>\listoffigures</code>	x	x	x	
<code>\listoftables</code>	x	x	x	
<code>\thebibliography</code>	x	x	x	
<code>\theindex</code>	x	x	x	
<code>\abstract</code>	x	x	x	
<code>\part</code>	x	x	x	
<code>\chapter</code>		x	x	
<code>\appendix</code>		x	x	
<code>\cc</code>				x
<code>\encl</code>				x
<code>\ps@headings</code>				x

Table 1: macros that need to be redefined for the four standard document styles.

```

\@ifundefined{contentsname}
  {\def\tableofcontents
    {\section*{\contentsname
      \@mkboth{\uppercase\expandafter{\contentsname}}
              {\uppercase\expandafter{\contentsname}}
      }
    \@starttoc{toc}
  }
}

```

Figure 6: An example of redefining a command

The macro `\extrasdutch` has a counterpart, `\noextrasdutch`, that cancels the extra definitions made by `\extrasdutch`. It changes the `\catcode` of the " character back to 'other' and globally `\lets` the macros `\dospecials` and `\@sanitize` to their original definitions. The original definition of " is restored as well.

In figure 10 the code needed to redefine `\dospecials` and `\@makeother` is shown.

8.3 An extra active character

All the code discussed so far is necessary because we need an extra active character. This character is then used as indicated in table 2. One of the reasons for this

is that in the Dutch language a word with an umlaut can be hyphenated just before the letter with the umlaut, but the umlaut has to disappear if the word is broken between the previous letter and the accented letter.

In [3] the quoting conventions for the Dutch language are discussed. The preferred convention is the single-quote Anglo-American convention, i.e. 'This is a quote'. An alternative is the slightly old-fashioned Dutch method with initial double quotes lowered to the baseline, „This is a quote”, which should be typed as " `This is a quote" '.

```

\@ifundefined{doc@style}
  {\def\doc@style{0}
    \@ifundefined{opening}
      {\@ifundefined{chapter}
        {\def\doc@style{1}}
        {\def\doc@style{2}}
      }{\def\doc@style{3}}
  }{\relax}

```

Figure 7: Determining the main document style


```

{\def\format{plain}
\ifx\fmtname\format
  \expandafter\ifx\csname @ifundefined\endcsname\relax
  \gdef\next{latexhax.sty}
  \aftergroup\input\aftergroup\next
\fi
\fi}

```

Figure 8: Conditional loading of latexhax.sty

```

\def\captionsdutch{\gdef\refname{Referenties}%
  \gdef\abstractname{Samenvatting}%
  \gdef\bibname{Bibliografie}%
  ...
  \gdef\pagename{Pagina}}

\def\datedutch{%
  \gdef\today{\number\day~\ifcase\month\or
    januari\or februari\or maart\or april\or
    mei\or juni\or juli\or augustus\or
    september\or oktober\or november\or december\fi
    \space \number\year}}

\begingroup \catcode'\''\active

\gdef\extrasdutch{%
  \global\let\dospecials\dutch@dospecials
  \global\let@sanitize\dutch@sanitize
  \catcode'\''\active
  \gdef"\{\protect\dutch@active@dq}
  \gdef"\{\protect@umlaut}
}\endgroup

\def\noextrasdutch{%
  \catcode'\''12
  \global\let\dospecials\original@dospecials
  \global\let@sanitize\original@sanitize
  \global\let\''\dieresis
}

```

Figure 9: The macros needed to switch to the Dutch language

8.3.1 Supporting macro definitions

The definition of the active " character needs a couple of support macros. The macro `\allowhyphens` is used make hyphenation of word possible where it otherwise would be inhibited by \TeX . Basically its definition is nothing more than `\nobreak \hskip 0pt plus 0pt`.

```

\gdef\allowhyphens{\penalty\@M
  \hskip\z@skip}

```

Then a macro is defined to lower the Dutch left double quote to the same level as the comma. It prepares a low double opening quote in box register 0. This macro was copied from `german.tex`.

```

\gdef\set@low@box#1{%
  \setbox\tw@\hbox{,}
  \setbox\z@\hbox{#1}
  \dimen\z@\ht\z@
  \advance\dimen\z@ -\ht\tw@
  \setbox\z@\hbox{\lower\dimen\z@
    \box\z@}
  \ht\z@\ht\tw@ \dp\z@\dp\tw@}

```

The macro `\set@low@box` is used to define low opening quotes. Since it may be used in arguments to other macros it needs to be protected.

```

\gdef\dlqq{\protect\@dlqq}
\gdef\@dlqq{%
  \ifhmode
    \edef\@SF{\spacefactor%
      \the\spacefactor}
  \else
    \let\@SF\empty
  \fi
  \leavevmode\set@low@box{' '}
  \box\z@\kern-.04em\allowhyphens\@SF\relax}

```

For reasons of symmetry we also define " '. This command is defined similar to `\dlqq`, except that the quotes aren't lowered to the baseline.

```

\gdef\drqq{%
  \ifhmode
    \edef\@SF{\spacefactor\the\spacefactor}
  \else
    \let\@SF\empty

```

```

\begingroup
  \def\do{\noexpand\do\noexpand}%
  \xdef\dutch@dosppecials{\dosppecials\do\}%
  \expandafter\ifx\csname @sanitize\endcsname\relax
% do nothing if \@sanitize is undefined...
  \else
    \def\@makeother{\noexpand\@makeother\noexpand}%
    \xdef\dutch@sanitize{\@sanitize\@makeother\}%
  \fi
\endgroup

\global\let\original@dosppecials\dosppecials
\global\let\original@sanitize\@sanitize

```

Figure 10: Code needed for the redefinition of `\dosppecials` and `\@makeother`.

```

"a \ "a which hyphenates as -a; also implemented for the
  other letters.
"| disable ligature at this position.
"- an explicit hyphen sign, allowing hyphenation in the rest
  of the word.
" ` lowered double left quotes (see example below).
" ' normal double right quotes.
\ - like the old \-, but allowing hyphenation in the rest of
  the word.

```

Table 2: The extra definitions made by `dutch.sty`

```

\fi
''\@SF\relax}}
The original double quote character is saved in the macro
\dq to keep it available.
\begingroup \catcode\`"12
  \gdef\dq{"}
\endgroup
The original definition of \` is stored as \dieresis.
The reason for this is that if a font with a different encoding
scheme is used the definition of \` might not be the plain TEX one.
\global\let\dieresis\`
In the Dutch language vowels with a dieresis or umlaut
accent are treated specially. If a hyphenation occurs before
a vowel-plus-umlaut, the umlaut should disappear. To be able
to do this, the hyphenation break behaviour for the five vowels,
both lowercase and uppercase, could be defined first in terms of
\discretionary. But this results in a large \if-construct in the
definition of the active ".
As both Knuth and Lamport have pointed out, a user should
not use " when he really means something like ' '. For this reason
no distinction is made between vowels and consonants. Therefore one
macro, \@umlaut, specifies the hyphenation break behaviour for all
letters.

\def\@umlaut#1{%
  \allowhyphens%
  \discretionary{-}{#1}{\dieresis #1}%
  \allowhyphens}
The last support macro to be defined is
\dutch@active@dq.

```

```

\gdef\dutch@active@dq#1{%
  \if\string#1'\dlqq{}%
\else\if\string#1'\drqq{}%
\else\if\string#1-\allowhyphens-%
  \allowhyphens%
\else\if\string#1|\discretionary{-}{}{%
  \kern.03em}%
\else\if\string#1i\allowhyphens%
  \discretionary{-}{i}{%
  \dieresis i}%
  \allowhyphens%
\else\if\string#1j\allowhyphens%
  \discretionary{-}{j}{%
  \dieresis j}%
  \allowhyphens%
\else \@umlaut{#1}\fi\fi\fi\fi\fi\fi}

```

The macro reads the next token and performs some appropriate action. If no special action is defined, it will produce an umlaut accent on top of argument 1.

The last definition needed is a replacement for `\-`. The new version of `\-` should indicate an extra hyphenation position, while allowing other hyphenation positions to be generated automatically. The standard behaviour of T_EX in this respect is very unfortunate for languages such as Dutch and German, where long compound words are quite normal and all one needs is a means to indicate an extra hyphenation position on top of the ones that T_EX can generate from the hyphenation patterns.

```

\def\-\{\allowhyphens\discretionary{-}{}{}%
  \allowhyphens}

```

8.4 Activating the definitions

The last action that should be performed by a language specific file, is activating it's definitions. Before doing that the macro `\originalTeX` should be defined.

```
\@ifundefined{originalTeX}{\let%
  \originalTeX\relax}{}
```

Also, the macro `\l@<language>` should be defined. If it hasn't already been defined, this means that no hyphenation patterns were loaded for this language.

```
\@ifundefined{l@dutch}{%
  \addlanguage{dutch}}{}
\selectlanguage{dutch}
```

9 Conclusion

In this article a system of document-style option files has been presented that support the multilingual use of \LaTeX . Some of the code involved has been discussed. The actual files will be made available through the international networks. They will be stored in the fileservers in the Netherlands (address: `LISTSERV@HEARN`), the file `babel_readme` will explain what you need to get to be able to use the system. The system was developed using the `doc` option, so the files available are fully documented.

References

- [1] Donald E. Knuth, *The \TeX book*, Addison-Wesley, 1986.
- [2] Leslie Lamport, *\LaTeX , A document preparation System*, Addison-Wesley, 1986.
- [3] K.F. Treebus. *Tekstwijzer, een gids voor het grafisch verwerken van tekst*. SDU Uitgeverij ('s-Gravenhage, 1988). A Dutch book on layout design and typography.
- [4] Hubert Partl, *German \TeX , TUGboat 9* (1988) #1, p. 70–72.
- [5] Leslie Lamport, in: \TeX hax Digest, Volume 89, #13, 17 februari 1989.
- [6] Frank Mittelbach, *The doc-option, TUGboat 10* (1989) #2, p. 245–273.
- [7] Johannes Braams, Victor Eijkhout and Nico Poppe-lier, *The development of national \LaTeX styles, TUGboat 10* (1989) #3, p. 401–406.
- [8] Joachim Schrod, *International \LaTeX is ready to use, TUGboat 11* (1990) #1, p. 87–90.

BIJLAGE T**Frequently Asked Questions¹****Version 1.13****Bobby Bodenheimer**

bobby@caltech.bitnet

March, 1991

Frequently Asked Questions

This article² contains answers to some frequently asked questions on comp.text.tex. Please don't ask these questions again, as they've been answered many times before.

This article includes answers to:

1. Where can I get a DVI to PostScript conversion program?
2. How can I include a PostScript figure in \LaTeX ?
3. Where can I find a DVI previewer for machine Y running Q?
4. Where can I get the manual for ΠCTeX ?
5. What is VorTeX and where can I get it?
6. What is OzTeX and where can I get it (TeX for the Mac)?
7. What is Fig and where can I get it?
8. How do I get WEB for C, FORTRAN, or some other language?
9. How can I typeset music in TeX ?
10. What is TUG and TUGboat?
11. How do I convert Adobe's afm files to tfm format?
12. In \LaTeX , how do I get a double-spaced document?
13. In \LaTeX , how do I include a file in the verbatim environment?
14. In \LaTeX , how do I do Y?
15. Where can I find a \LaTeX style file for doing Y?
16. How do I generate an index in TeX/\LaTeX ?
17. How do I get METAFONT to do what I want it to do?
18. Where do I get TeX/\LaTeX for machine Y running Q?
19. What is the Clarkson archive server?
20. Where can I get a thesis style for \LaTeX ?
21. How do I get symbols for "the real numbers", "the complex numbers", and so on?
22. How do I get TeX material if I do not have access to anonymous ftp?
23. How do I use PostScript fonts with \LaTeX ?
24. How can I convert from format Y to TeX or \LaTeX , and vice-versa?
25. How do I get a file into the major style repositories?
26. Where can I get font Y?
27. Where can I get a dvi driver for the HP LaserJet?

These are all legitimate questions, but they seem to appear too frequently for long-time readers of the list.

Many of the answers below tell you that you can obtain something through anonymous ftp. Remember that anonymous ftp is a privilege and that the system administrators for these sites have made these files available out of their own generosity. Therefore please restrict your ftp'ing to non-prime hours at the various sites. Also note that Raymond Chen (raymond@math.berkeley.edu) posts a monthly document "Supplementary TeX Information" to this newsgroup which contains other information and software relevant to TeX users but beyond the scope of this article. A \LaTeX version of this article is available via anonymous ftp from ymir.claremont.edu (134.173.4.23) in [anonymous.tex.documentation] and from the other inclusive archives. The most recent version should be available no later than a week from the date of this pos-

¹This article has been adapted by Piet van Oostrum in order to reflect the local situation.

²Deze bijlage bevat een aangepaste versie van de "Frequently asked questions" uit de Usenet nieuwsgroep "comp.text.tex". De aanpassingen bestaan erin dat ftp adressen vervangen zijn door referenties naar de file servers op RUU en HEARN waar mogelijk. De originele tekst en de gerefereerde "Supplementary TeX information" gezamenlijk zijn te verkrijgen op RUU als FAQ/comp.text.tex.

Als in deze file staat: op RUU te verkrijgen als x/y/z (of equivalent: als z in directory x/y), dan kunt u:

- Via ftp archive.cs.ruu.nl [131.211.80.5] de file ophalen als pub/x/y/z.
- Via mail een boodschap sturen naar: mail-server@cs.ruu.nl met als inhoud: send x/y/z

Als u de mail-server niet kent, stuur dan eerst een boodschap met als inhoud: HELP

Als uw mail-verzender niet algemeen geldige adressen genereert dan is het verstandig om een regel op te nemen met als inhoud het woord PATH gevolgd door een adres in internet stijl. Voor gebruikers op BITNET is dat: PATH user@host.bitnet

De files op HEARN zijn te verkrijgen door een boodschap te sturen naar LISTSERV@HEARN.bitnet.

ting.

I would like to acknowledge Don Hosek, Ken Yap, Tomas Rokicki, and Micah Beck, whose postings provided many of the answers. Joe Weening, Hal Perkins, Walter Carlip, Max Hailpern, Tad Guy, Raymond Chen, Henning Schulzrinne, Sebastian Rahtz, Mark James, Peter Galko, and Mike Ernst provided additional material and criticisms. The format of this document is based on that of the Frequently Asked Questions appearing in comp.unix.wizards, and written by Steve Hayman. Any mistakes are mine. Corrections, suggestions, and additions to bobby@hot.caltech.edu.

1 Where can I get a DVI to PostScript conversion program?

Four very nice DVI to PostScript conversion programs that run under Unix are:

- **dvitps** part of the \TeX XPS package by Stephan Bectolsheim.
Available via anonymous ftp from arthur.cs.purdue.edu (128.10.2.1) in ./pub/TeXPS.
- **dvi3ps** by Kevin Coombes.
Available on RUU in TEX/DVI/dvi3ps.tar.Z
- **dvips** by Tomas Rokicki.
This driver is very nice and has the ability to deal with virtual fonts. Available on RUU in TEX/DVI/dvips547.tar.Z, dvipslib.tar.Z, dvipsafm.tar.Z. Dvips ports easily to other operating systems. It is available for VMS via anonymous ftp from ymir.claremont.edu (134.173.4.23) in [.tex.drivers.dvips_new] and also through the DECUS library (see question 22). A precompiled version for MSDOS is available from RUU in TEX/DVI/dvips547_msdos_exes.zip. If you wish to use postscript fonts, get dvipslib.zip as well.
- **dvitops** by James Clark.
Available via anonymous ftp from june.cs.washington.edu (128.95.1.4) in ./tex/dvitops.tar.Z and ymir.claremont.edu (134.173.4.23) in [anonymous.tex.drivers.dvitops]. Dvitops will compile under Unix, MSDOS, VMS, and Primos.

2 How can I include a PostScript figure in \LaTeX ?

Perhaps the best way to do this is to use the psfig macros written by Trevor Darrell. They are available via anonymous ftp from whitechapel.media.mit.edu (18.85.0.125) in ./psfig or linc.cis.upenn.edu (130.91.6.8) in the directory ./dist/psfig. You will also need a dvi to PostScript conversion program that supports \specials. The ones mentioned in question 1 do, and the first two drivers come with a version of psfig ready to use with them. The psfig macros work best with Encapsulated

PostScript Files (EPS). In particular, psfig will need the file to have a BoundingBox (see Appendix C of the PostScript Language Reference Manual). If you don't have an EPS file, life can be difficult. For people who don't have ftp access or can't deal with tar files, the files are also available from ymir.claremont.edu (134.173.4.23) in [anonymous.tex.graphics.psfig].

3 Where can I find a DVI previewer for machine Y running Q?

This briefly lists some previewers which are available via anonymous ftp:

- **dvipage** For SunView. Available via anonymous ftp from june.cs.washington.edu (128.95.1.4) in ./tex/dvipages.tar.Z.
- **xtex** For X Windows. Available via anonymous ftp from foobar.colorado.edu (128.138.243.105) in ./pub/SeeTeX/SeeTeX/SeeTeX-2.17.x.tar.Z.
- **dviapollo** For Apollo Domain. Available via anonymous ftp from june.cs.washington.edu in ./tex/dviapollo.tar.Z.
- **dviis** For Integrated Systems. Available via anonymous ftp from june.cs.washington.edu in ./tex/dviis.tar.Z.
- **dvidis** For VAXstation VWS. Available via anonymous ftp from venus.ycc.yale.edu (130.132.1.5) in [.dvidis] (VMS machine).
- **xdvi** Also for X Windows. Available on RUU in TEX/DVI/xdvi.shar.
- **dvitovdu** For Tektronix 4010 and other terminals under Unix. Available via anonymous ftp from wsmr-simtel20.army.mil (26.2.0.74) in the directory pd2:<unix-c.printers> as dvi2vdu.tar-z (ftp in "tenex" mode). A C version is also available from ymir.claremont.edu (134.173.4.23) in [anonymous.tex.drivers.dvitovdu.c-1].
- **dvi2tty** A dvi to ASCII conversion program, for normal terminals. Available from RUU in /TEX/DVI/dvi2tty.shar. I could not get this program to run on a Sun running SunOS 4.1.

4 Where can I get the manual for $\Pi\text{CT}_{\text{E}}\text{X}$?

The $\Pi\text{CT}_{\text{E}}\text{X}$ manual is not free. It is available for \$30 (\$35 with the disk) from the \TeX X Users Group:

\TeX X Users Group
P. O. Box 9506
Providence, RI 02940 (USA)
401-751-7760

tug@math.ams.com

The proceeds from this sale go to Michael Wichura, the author of PiCT_EX, and TUG.

5 What is VorT_EX and where can I get it?

VorT_EX is a package of programs written at the University of California. It includes several nice previewers and some Emacs modes for T_EX and BibT_EX. It is not free. Inquiries should be directed to

vortex@ucbarpa.berkeley.edu

or

Professor Michael A. Harrison
Att. Vortex Dist.
Computer Science Division
University of California
Berkeley, CA 94720

6 What is OzT_EX and where can I get it (T_EX for the Mac)?

OzT_EX is a public domain version of T_EX for the Macintosh. A DVI Previewer and PostScript driver are also included. It should run on any Macintosh Plus, SE, II, or newer model, but will not work on a 128K or 512K Mac. It was written by Andrew Trevor-row, and is available via anonymous ftp from from midway.uchicago.edu (128.135.12.73) in ./pub/OzTeX, which contains other public domain T_EX-related software for the Mac as well. Questions about OzT_EX may be directed to oztex@midway.uchicago.edu.

7 What is Fig and where can I get it?

Fig is a menu driven tool similar to MacDraw that allows you to draw objects on the screen of a Sun Workstation running SunView. TransFig is a set of tools which translate the code fig produces to other graphics languages including PostScript and the L^AT_EX picture environment. Both are available via anonymous ftp from svax.cs.cornell.edu (128.84.254.2) in ./pub/fig. Both Fig and TransFig are also available from the Clarkson archive server at sun.soe.clarkson.edu (see question 19). Both Fig and TransFig are supported by Micah Beck (beck@svax.cs.cornell.edu).

XFig is essentially the same program except that it runs under X Windows. It is available via anonymous ftp from expo.lcs.mit.edu (18.30.0.212) in ./contrib/xfig-2.0.*.Z. Note that version 2.0 is the most recent. It was written by Brian Smith.

For complete compatibility with TransFig, be sure to get at least patchlevel 4 of XFig 2.0.

8 How do I get WEB for C, FORTRAN, or some other language?

There is a version of WEB for C called CWEB written by Silvio Levy. It is available on RUU in the directory TEX/TOOLS/cweb.

There is a version of WEB called Spidery WEB which supports many languages including ADA, awk, and C. It was written by Norman Ramsey and, while not in the public domain, is usable free for research purposes. It is available on RUU in TEX/TOOLS/spiderweb.tar.Z.

There is a version of WEB called FWEB for Fortran, Ratfor, and C written by John Krommes (krommes@lyman.pppl.gov). Version 1.13 is available via anonymous ftp from lyman.pppl.gov (192.55.106.129) in ./pub/fweb.

SchemeWEB is a Unix filter that translates SchemeWEB into L^AT_EX source or Scheme source. It was written by John Ramsdell and is available from the Clarkson archive (see question 19) in ./submit/schemeweb.sh.

All of the above are also available from ymir.claremont.edu (134.173.4.23) in [anonymous.tex.utilities].

9 How can I typeset music in T_EX?

A package called MuT_EX, written by Andrea Steinbach and Angelika Schofer, aids in doing this. It is available on RUU in TEX/MuTeX.tar.Z. This package allows you to typeset single-staff music and lyrics.

A more powerful package which allows the typesetting of orchestral and polyphonic music is MusicT_EX, written by Daniel Taupin. It is also available on RUU as TEX/musictex.tar.Z

10 What is TUG and TUGboat?

TUG is the T_EX User's Group. TUGboat is their newsletter, which publishes useful articles about T_EX and METAFONT. Inquiries should be directed to:

T_EX Users Group
P. O. Box 9506
Providence, RI 02940 (USA)
401-751-7760
tug@math.ams.com

11 How do I convert Adobe's afm files to tfm format?

Use the afm2tfm program distributed with dvips (see Question 1)

For the Macintosh, there is a program called EdMetrics which does the job (and more).

It is available free from:

Blue Sky Research
534 Southwest Third Avenue
Portland, Oregon 97204 (USA)
800-622-8398 or 503-222-9571

12 In \LaTeX , how do I get a double-spaced document?

One way is to change the value of `\baselinestretch` to 2 or 1.5 with the command `\renewcommand{\baselinestretch}{2}`. A better way is to use the `doublespace` style file `doublespace.sty` on RUU in `TEX/latexstyle/` or, if you are using the new font selection scheme, get the `doublespace.sty` from `yimir.claremont.edu` (134.173.4.23) in `[anonymous.tex.inputs.latex-contrib]`.

13 In \LaTeX , how do I include a file in the verbatim environment?

A good way to do this is to use Rainer Schoepf's `verbatim.sty`, which provides the command `\verbatiminput` that takes a file as an argument. This file is available on RUU in `TEX/latexstyle/verbatim.zoo`

Another way to do this is to use the `alltt` environment defined in the style file `alltt.sty` available in `TEX/latexstyle/` on RUU.

14 In \LaTeX , how do I do Y?

If you can't figure out how to do something in \LaTeX after you have read the manual very carefully, asked your local \LaTeX guru, and thought about it, there is a \LaTeX help service available. Please note that the way to accomplish something in \LaTeX is often by using an appropriate style file, so please check this also (see question 15). If none of this works, send mail describing your problem to `latex-help@cs.stanford.edu`. If you haven't gotten a reply to your problem within about a week, send mail to `latex-help-coordinator@cs.stanford.edu`.

15 Where can I find a \LaTeX style file for doing Y?

Before you ask for a \LaTeX style file to do something, please check the \LaTeX style collection at RUU in `TEX/latexstyle`. It should be noted that the basic \LaTeX style files are available on RUU in the directory `TEX/TEX3/latex/` and on HEARN as `LATEXSRC UUE`, while `latexstyle` contains only supplementary style files.

16 How do I generate an index in \TeX/\LaTeX ?

Making an index is not trivial. There are several indexing programs which aid in doing this. Some are:

- **makeindex**
For \LaTeX under Unix (but runs under other OS's without changes). Available on RUU in `TEX/TOOLS/makeidx`. A version for the Macintosh is available from Johnny Tolliver at `tolliver%atf.mfenet@nmfecc.llnl.gov`.
- **idxtex**
For \LaTeX under VMS. Available via anonymous ftp from `yimir.claremont.edu` in the directory `[.tex.utilities.idxtex]`.
- **texix**
For \TeX on CMS and Macintosh machines. Available via anonymous ftp from `yimir.claremont.edu` in the directory `[.tex.utilities.texix]`.
- **indexor**
For \LaTeX under Unix, VMS, and DOS. Available via anonymous ftp from `yimir.claremont.edu` in `[.tex.utilities.indexor]`.

17 How do I get METAFONT to do what I want it to do?

METAFONT allows you to create your own fonts, and ordinary \TeX users will never need to use it. METAFONT, unlike \TeX , requires some customization. Each output device for which you will be generating fonts needs a mode associated with it. Modes are defined using the `mode_def` convention described on page 94 of 'The METAFONTbook'. So first create a file, which we will call `local.mf`, containing all the `mode_defs` you will be using. The file `waits.mf` (in the standard distribution or available on RUU in `TEX/TEX3/lib`) is a good starting point for this. Listings of settings for various output devices are also published periodically in TUGboat (see question 10). Now create a plain base file using `inimf`, `plain.mf`, and `local.mf`:

```
% inimf
This is METAFONT....
**plain          # you type plain
(output)
*input local     # you type this
(output)
*dump            # you type this
Beginning to dump on file plain....
(output)
%
```

This should create a base file named `plain.base` (or something close) and should be moved to the directory which contains the base files on your system.

Now we need to make sure that METAFONT loads this base when it starts up. If METAFONT loads the plain

base by default on your system, then you're ready to go. Under Unix, we might, for instance define a command `mf` which executes "`virmf &plain,`" loading the plain base file.

The usual way to create a font with plain METAFONT is to then start it with the line

```
\mode=<mode name>;
mag=<magnification>;
input <font file name>
```

in response to the `*` prompt or on the METAFONT command line. If `<mode name>` is unknown or omitted, then the mode defaults to proof mode. If this has happened METAFONT will produce an output file called `.2602gf`. The `<magnification>` is a floating point number or magstep (magsteps are defined in 'The METAFONTbook' and 'The T_EXbook'). If `mag=<magnification>` is omitted, then the default is 1. For example, to generate `cmr10` at 12pt for an epson printer you would type

```
\mode=epson; mag=1.2; input cmr10
```

Note that under Unix the `'\'` and `';` characters must usually be escaped, so that this would typically look something like

```
mf \\mode=epson\; mag=1.2\; input cmr10
```

If you don't have `inimf` or need a special mode that isn't in the base, you can put its commands in a file (e.g., `ln03.mf`) and invoke it on the fly with the `\smode` command. For example, to create `ln03.300gf` for an LN03 printer, using the file

```
% This is ln03.mf as of 2/27/90
% mode_def courtesy of John Sauter
proofing:=0;
fontmaking:=1;
tracingtitles:=0;
pixels_per_inch:=300;
blacker:=0.65;
fillin:=-0.1;
o_correction:=.5;
  (note the absence of the mode_def
  and enddef commands), we would type
mf \smode="ln03"; input cmr10
```

18 Where do I get T_EX/L^AT_EX for machine Y running Q?

- **Unix**

The Unix T_EX distribution is available for anonymous ftp from `labrea.stanford.edu` (36.8.0.47) in the directory `./tex/unix`. It is large, so please be sure to observe proper ftp etiquette when getting it. It is also available through the University of Washington, for a small fee. Contact:

Director

Northwest Computing Support Center
Thomson Hall, Mail Stop DR-10
University of Washington
Seattle, WA 98195 (USA)
(206)543-6259

or send electronic mail to Elizabeth Tachikawa at `elisabet@max.u.washington.edu` (note the "s"). The fee charged for getting the T_EX distribution through the University of Washington helps fund the further development of Unix T_EX, so it's a good idea to order it this way. This distribution compiles under Ultrix. Executables for the 386/ix are available via anonymous ftp from `tik.vtt.fi` (130.188.52.2) in `./pub/tex/bin-386ix` and from `math.berkeley.edu` (128.32.183.94) in `./pub/tex386ix.tar.Z`.

- **PC**

A T_EX package for the PC, including L^AT_EX, BibT_EX, previewers, and drivers is available via anonymous ftp from `vax.eedsp.gatech.edu` (130.207.226.2) in `./pub/TeX`. The variety here is sbT_EX version 30 by Wayne Sullivan. EmT_EX, another T_EX package for the PC by Eberhard Mattes, is available on RUU in `TEX/emtex`. This package includes L^AT_EX, METAFONT, BibT_EX, etc., as well. Documentation is available in both German and English.

All Public Domain T_EX software for the PC is also available through

Jon Radel
P. O. Box 2276
Reston, VA 22090
`jonradel@phoenix.princeton.edu`

Send a self-addressed stamped envelope with 10 sheets of paper or \$2.00 to obtain information on what's available.

- **Mac**

See question 6 for a public domain version.

- **TOPS-20**

T_EX was originally written on a DEC-10 under WAITS, and so was easily ported to TOPS-20. A Distribution that runs on TOPS-20 is available via anonymous ftp from `science.utah.edu` (128.110.198.2) in `./pub/tex/pub/web`.

- **VAX/VMS**

VMS executables are available via anonymous ftp from `yimir.claremont.edu` (134.173.4.23) in `[.tex.exe]`. Source is available in `[.tex.sources]`. Version 3.1 is available in `[.tex.sources.tex3_1]`. Yimir has a mailserver for those without ftp access, although executables are not available through it. Send a message containing the line "help" to `mailserv@yimir.claremont.edu`. Standard tape distribution is through DECUS or Maria Code.

- **Atari**

T_EX is available for the Atari ST on RUU in `ATARI-ST/tex/`

- **Amiga**

Disk 1 of the Amiga CommonT_EX distribution is available via anonymous ftp from `ab20.larc.nasa.gov` (128.155.23.64) in `./amiga/commontex.lzh`. The file

copying.tex in the distribution contains information on how to obtain the other disks. Also, experimental versions of \TeX 3.1 and METAFONT 2.7 are available in ./incoming/amiga/TEX.

- **Tandy 6000**

First get web2c-5.8a and web-5.8a from ics.uci.edu (128.195.1.1) in ./TeX and then apply the diffs available and plains.ndsu.nodak.edu (134.129.111.64) in ./pub/tandy/programming and build. This builds \TeX 3.1 and METAFONT 2.7.

19 What is the Clarkson archive server?

The Clarkson archive server is a program that allows you to retrieve files via electronic mail. So it provides a way of getting many neat things if you don't have anonymous ftp. To get the archive server to do something you should send a message to archive-server@sun.soe.clarkson.edu (128.153.12.3). To get started using it, send the archive server a one line message "help" (case is unimportant). It will send you back a file describing how to use it more fully.

In particular, the Clarkson archive contains, among other things, a repository for \LaTeX style files, \AMSTeX macros and style files, \BibTeX style files, \PiCTeX source (not the manual, see question 4), back issues of \TeX hax and \TeX MaG, files from TUGboat, and source to \TeX , \LaTeX , WEB, and various utilities. The archive is maintained by Michael DeCorte.

If anonymous ftp is available to you, you may get the files directly from sun.soe.clarkson.edu.

If you have problems, contact archive-management@sun.soe.clarkson.edu.

Ymir.claremont.edu (134.173.4.23) is also a \TeX repository with lots of other stuff as well. It can be reached with anonymous ftp and also has a mailing program for people without anonymous ftp. See question 22 for more information.

20 Where can I get a thesis style for \LaTeX ?

Thesis styles are usually very specific to your University, so it's usually not profitable to ask the whole newsgroup for one. If you want to write your own, a good place to start is the ucthis style available in the \LaTeX style collection

(see questions 15 and 19).

21 How do I get symbols for "the real numbers", "the complex numbers", and so on?

These symbols are known as "blackboard bold" and are available in the recently released AMS fonts "msam" (e.g., "msam10" for 10pt) and "msbm". They replace the older "msxm" and "msym." The fonts have a large number of mathematical symbols to supplement the ones provided by \TeX . The fonts are available on RUU in the directory TEX/AMS/amsfonts and on HEARN as AMS-FONTS UU1 ... AMSFONTS UUA Two files which load the fonts and define the symbols are provided, and both work with either \TeX or \LaTeX . Additionally a replacement for the old amssymbols.sty file using the new fonts was posted by Tony Li on August 13, 1990, but does not seem to have made it into the major style repositories. Questions or suggestions regarding these fonts should be directed to tech-support@math.ams.com.

22 How do I get \TeX material if I do not have access to anonymous ftp?

There are several mail servers available, i.e., programs that will send you files via electronic mail. Some are:

- The mail server at Clarkson (see question 19), if you can send mail to the United States.
- Also in the US, the mail server at ymir.claremont.edu has access to all the \TeX -related material there. This site has a lot of the PD \TeX software, and even if it's not explicitly stated, much of the software mentioned in this document is available there. Unfortunately, executables are not available through the mail server. Send a message containing the line "help" to MAILSERV@ymir.claremont.edu.
- In the United Kingdom, you can get \TeX -related material from the UK \TeX archive at the University of Aston. (FTP access is also available for people on JANET:

```
Site uk.ac.tex
Username public
Password public
Top Level tex-archive
```

A known file is [tex-archive]000directory.list.) For electronic mail access, send a message to texserver@uk.ac.tex. The first non-blank line of the message must contain a valid \TeX server command (help, directory, files, whereis, search, or path). The program will then mail you a response notifying you that your request has been received. If you fail to get a response from the \TeX server, you may need to use the path command to help the program out. For Internet users the return address is of the form name%site@nsfnet-relay, while for Bitnet and EARN it is name%site@earn-relay (i.e., include a line that says "path name%site@nsfnet-relay along with a line

containing "help"). Note that the old three hyphen format is obsolete, but still accepted by the program for backward compatibility.

- archive.cs.ruu.nl (131.211.80.5) also contains a substantial T_EX archive (ftp access is available). Send a messages containing the line "help" to mail-server@cs.ruu.nl. This mail server can send binary files in a variety of different formats.
- There are LISTSERV facilities for T_EX at LISTSERV@DHDURZ1.BITNET. Send a message containing the line "help" to this address.
- For users on BITNET, access to anonymous ftp for some files can be obtained indirectly by sending mail to BITFTP@PUCC.BITNET. Send a message containing the line "help" to this address for more information.

There is also the DECUS T_EX collection, a collection of T_EX material for VMS, Unix, MS-DOS, and the Macintosh. It can be obtained from the DECUS Library (reference number VS0058) in the US, or through your DECUS office outside of the US. To contact the DECUS Library, send mail or call:

The DECUS Program Library
219 Boston Post Road BP02
Marlboro, MA 01752-1850
(508)480-3418

or send electronic mail to the DECUS T_EX Collection Editor, Ted Nieland, at decus_tex@nieland.dayton.oh.us.

Another good source of information is NETWORK SOURCES OF T_EX WARE by Peter Flynn which appeared in T_EXhax, volume 90, issues 45-47 (in May 1990).

23 How do I use PostScript fonts with L^AT_EX?

There are at least four problems with replacing the standard Computer Modern fonts used in L^AT_EX:

1. The fonts are defined in a long set of macros in lfonts.tex which are not easy to understand and change;
2. You must have a device driver capable of understanding the idea of fonts built into the printer (all the drivers referred to in question 1 have this capability).
3. The default layout of the Adobe fonts is NOT the same as that for the CMR fonts, so you will not get the right characters (ligatures being a good example).
4. The official names of the PostScript fonts are long and in both upper and lower cases. While this presents no problems for Unix, it can cause problems with files under DOS, VMS, etc.

The first problem can be solved by one of:

1. Going through lfonts.tex and changing references to CMR to some new font, e.g., times.

2. Overloading the definitions of \bf, \rm, etc., with new font families.
3. Using the font selection mechanism described in TUGboat 10.3, p. 222 ff; a L^AT_EX interface to this will be part of version 2.10 of L^AT_EX, and is currently available. From RUU get the file TEX/latexstyle/fontsels.zoo.

The second problem is quickly disappearing since all recent DVI to PostScript drivers have had adequate facilities. The third problem can be solved by:

1. Changing T_EX macros, etc., so that the Adobe font layouts are expected.
2. Re-encoding the Adobe fonts in a PostScript prolog so that they conform to T_EX standards.
3. Using "virtual fonts" to perform the mapping between Adobe and T_EX layouts. Tomas Rokicki's dvips driver (see question 1) has examples of this.

A definitive statement on short names for Adobe fonts is expected soon. Meanwhile, there is not much agreement on how to compress Garamond-LightItalic into 8 characters.

24 How can I convert from format Y to T_EX or L^AT_EX, and vice-versa?

- **troff**
troff-to-latex.tar.Z is available on RUU in TEX/TOOLS/troff-to-latex.tar.Z. This program, written by Kamal Al-Yahya at Stanford, assists in the translation of a troff document into L^AT_EX format. It recognizes most -ms and -man macros, plus most eqn and some tbl preprocessor commands. Anything fancier than that needs to be done by hand. Two style files are provided. There is also a man page (which converts very well to L^AT_EX :-). The program is copyrighted but free. The DECUS T_EX distribution (see question 22) also contains a program which converts troff to T_EX.
- **scribe**
Mark James has a copy of scribe2latex which he has been unable to test but which he will let anyone interested have. Send email to mark@bdbluess.altair.fr. The program was written by Van Jacobson of Lawrence Berkeley Laboratory.
- **wordperfect**
wp2latex.zip is available on RUU in TEX/TOOLS/wp2latex.arc and on HEARN as WP2LATEX.UUE. This is a PC program written in Turbo Pascal by R. C. Houtepen at the Eindhoven University in the Netherlands. It converts WordPerfect 5.0 documents to L^AT_EX. Pascal and (on RUU) C source is included. Users find it "helpful" and "decent" in spite of some limitations. It gets high marks for handling font changes. Limitations include no indices, table of contents, margins or graphics. It also

won't handle all the new features of WordPerfect 5.1, in particular the equation formatter. The program is copyrighted but free.

- **PC-Write**
pcwritex.arc is available on wuarchive.wustl.edu (128.252.135.4) in directory mirrors/msdos/tex and on wsmr-simtel20.army.mil (26.2.0.74) in directory pd1:<msdos.tex>. This is a print driver for PC-Write that "prints" a PC-Write V2.71 document to a T_EX-compatible disk file. It was written by Peter Flynn at University College, Cork, Ireland. It is public domain.
- **runoff**
Peter Vanroose, of the University of Leuven, Belgium, has written a RUNOFF-to-T_EX conversion program in VMS Pascal. It is available from comp.text archives (they do exist, don't they? The program was submitted in December 1987) or from the author (peter@dit.lth.se) or from Mark James (mark@bdbluess.altair.fr).
- **refer/Tib**
There are a few programs for converting bibliographic data between BibT_EX and refer/Tib formats. They are available via anonymous ftp from wheaties.ai.mit.edu (128.52.32.13) in the directory ./refer-to-bibtex. In spite of the directory name, it also contains a shell script to convert BibT_EX to REFER as well. This collection is maintained by Thomas M. Breuel (tmb@ai.mit.edu).

In addition, a group at Ohio State University is working on a common document format based on SGML. In theory any format could be translated to or from this one. Also, Framemaker supposedly has "import filters" to aid in the translation from alien formats (presumably including T_EX) to Framemaker; perhaps other desktop publishing programs have similar things.

25 How do I get a file into the major style repositories?

Use anonymous ftp to fenris.claremont.edu (134.173.4.146) and transfer the file into the directory ./pub/tex-incoming. Then send notification to tex-group@hmcvax.claremont.edu. If the file is large, it's better to send the note before transferring the file. From there it will propagate to other inclusive archives.

26 Where can I get font Y?

A comprehensive list of METAFONT fonts is posted to Comp.fonts about once every six weeks by Lee Quin (lee@sq.sq.com). It contains both commercial fonts and fonts available via anonymous ftp. Most of the fonts available via anonymous ftp are available from ymir.claremont.edu (134.173.4.23). Also, the file wujastyk.txh on ymir.claremont.edu in [anonymous.tex.mf] is a copy of Dominik Wujastyk's font article, and contains information on METAFONT fonts as well.

27 Where can I get a dvi driver for the HP LaserJet?

PC - The emT_EX package mentioned in question 18 contains a driver for the LaserJet, dvihplj. The driver is available by itself from RUU in TEX/emtex/disk3 in the files dvidrv[123].zip.

See also TEX/DVI/dvi2lj.arc on RUU.

Version 2.10 of the Beebe drivers support the LaserJet. These drivers will compile under Unix, VMS, and on the Atari ST and DEC-20's. They are available from science.utah.edu (128.110.198.2) in ./pub/tex/dvi and from ymir.claremont.edu (134.173.4.23) in [.tex.drivers.beebe2_10].

BIJLAGE U**Summary of Metafont Fonts Available¹****Liam R. E. Quin**SoftQuad Inc.,
Toronto,
+1 (416) 963-8337
lee@sq.com

December, 1990

This report includes all known fonts available in metafont format, whether public domain or not. Archive sites for ftp are listed where known. There is also a BITNET archive at LISTSERV@UBVM.CC.BUFFALO.EDU.

I also included some notes on how to use the fonts, in the (probably vain) hope of avoiding a deluge of questions on the net. Note that I cannot give instructions for specific packages, and I cannot answer questions about specific printers. I have included some information about T_EX and troff, but I can't go into more detail over the net (i.e. I am not offering to provide detailed technical support).

1 About Metafont

METAFONT is a programming language for describing fonts. It was written by Donald Knuth and is documented in *Computers & Typesetting/C: The METAFONT book* Knuth, Donald E. Addison Wesley, 1986 ISBN 0-201-13445-4, or 0-201-13444-6 (soft cover) Library access: Z250.8.M46K58, or 686.2'24, or 85-28675.

A font written in METAFONT is actually a computer program which, when run, will generate a bitmap ('raster') for a given typeface at a given size, for some particular device.

2 What you need in order to use the fonts

You cannot print the METAFONT files directly (unless you want a listing of the program, that is). Instead, you must generate a bitmap font and use that to print something. If you have T_EX or troff, the process for doing this is outlined below. This is not meant to be complete documentation, though, but simply enough to get you started.

If you want to use a font generated by METAFONT

on an HP LaserJet, for example, you must follow this general procedure:

- Generate the bitmap font with METAFONT. This will produce a file called (for example) "myfont10.300gf".
- Convert this to a pk format file – the pk format is much more compact, and is used by most software in preference to gf files. You can use gftopk (which comes with METAFONT, or in a package called mftware).

```
$ gftopk myfont10.300gf
$
```

This produces myfont10.300pk

- Convert the pk-format font into one suitable for your printer. For an HP, there are several utilities – the one I use is called pk2sfp:

```
$ pk2sfp myfont10.300pk >
myfont10.300sfp
```

You will have to find this program. If you are using T_EX you do not need it, though – see "Converting between font formats" below.

- Download the font to the printer. You will need to do more than simply send the file to the printer. For example, on the HP you must send an escape sequence like ESC*c100D to tell the printer there's a font coming, and that it's going to be font number 100. You then send the font, and then in the rest of the job say something like ESC*c100ESC*c4FESC (100X to tell the printer you've finished, and to start using the font. (ESC in these examples stands for the ASCII Escape, character 033 octal, 27 in decimal). All of this must be in the same print job.

Doing this without going insane involves setting up macros or shell-scripts to automate it for you.

3 How to use Metafont fonts with T_EX

In addition to generating a 'gf' file, METAFONT will also generate a '.tfm' file. You will need to put the

¹ Published in TeXMaG; Volume4, Number 6; dec 1990.

tfm file into your \TeX font directory – for example, /usr/local/tex/fonts. You will need to put either the gf or the pk file there too – probably the pk file. Some dvi drivers need the font to be listed in a description file, too – the name and location of this seems to vary wildly from site to site, but FONTDESC is not unlikely. Chris Torek's 'mctex' package includes some drivers that need this.

For testing, you might be able to say something like `$TEXFONTS="/users/lee/fonts:/usr/local/lib/tex/fonts";` export TEXFONTS to get \TeX to look in more than one directory – see your local \TeX guide.

When you have installed the font, you can use it from \TeX directly in the usual way. For most fonts, the filename is the same as the font name, so, for example, ccr10.300pk contains the bitmaps for a font called ccr at size 10pt. You can look at the METAFONT source for a given font to determine the name of the font, and you might also be able to cheat by editing the font description file FONTDESC.

Using other fonts with \LaTeX generally requires more work.

4 How to use Metafont fonts with Troff

If, when you run troff, you get the message 'typesetter busy', you have the original Ossanna-troff, also called otroff. Chris Lewis has a package which will let you use \TeX fonts with troff – it's called psroff, and comes with documentation.

ftp: gatekeeper.dec.com (16.1.0.2)
pub/misc/psroff-2.0.tar.Z
ftp: cs.toronto.edu [128.100.1.65]
pub/psroff.tar.Z

If, when you run troff, you get something like this:

```
x T 300
x res 300 1 1
```

you have ditroff. This is sometimes called titroff or psroff. In this case, you will probably need to do the following:

1. convert the font to your printer's format
2. generate a width table for the font
3. add the font to the DESC file for the appropriate device
4. arrange for troff to download the font
5. tell troff about the font by running 'makedev DESC' in the right place.

If, when you run troff, you get something like this:

```
X hp(SCM)(CM)(AF)(AD) 300 1 1
Y P default letter 2550 3300 0 0 90 90
2460 3210
```

you have sqtroff, change (4) onwards to:

4. put the font in the appropriate raster directory
5. tell sqtroff about the font by running 'sqmakedev DESC'

or 'sqinstall'.

In each case, you should be able to get help from your vendor.

Note that Chris Lewis' psroff package has software to make width tables for troff from pk files.

5 Converting between font formats

Conversions to and from pbm and pk format were posted to comp.text.tex and to alt.sources on the 9th of August, 1990 by Angus Duggan (ajcd@cs.ed.ac.uk). The program is pbmtopk, and there are also at least two patches.

Chris Lewis' psroff package includes a program to go from pk both to the HP LaserJet and to PostScript.

John McClain (ophelp@tamvenus.bitnet) has some conversion programs for various graphics formats to/and from pk files.

CAPTURE turns HPGL files into PK format (a PC program, \$130 from Micro Programs Inc., 251 Jackson Ave., Syosset, NY 11791

Metaplot can take pen-plotter files and prouce metafont files wilcox@cis.ohio-state.edu

6 Where to get bitmap versions of the fonts

There are archives containing the bitmaps of many of these fonts at various sizes and resolutions. The fonts must have been generated for the correct print engine: e.g. write-white or write-black. The archives generally hold only the sizes used by \TeX . These are 'magstep' sizes, and are not exact point sizes. It is probably better to generate them from the METAFONT sources yourself if you can.

The best place to look for raster fonts is almost certainly mims-iris.waterloo.edu (129.97.129.116)

This probably has all the HP fonts in the world.... some others are:

```
ctrsci.math.utah.edu (128.110.198.1)
science.utah.edu (128.110.192.2)
ymir.claremont.edu (134.173.4.23)
```

The occasional posting of ftp sites to comp.misc and comp.archives lists these and several other sites.

Getting fonts by FTP and Mail:

If you are using ftp, you will need either the name of the host or the Internet number. For example, to connect to ymir, listed as

ftp: ymir.claremont.edu [134.173.4.23] you will need to type something like

```
ftp ymir.claremont.edu
```

If that doesn't work, try using the number:

```
ftp 134.173.4.23
```

If that doesn't work, on Unix systems you can use `nslookup` (it's usually `/usr/etc/nslookup`) to find the host number – it might have changed. Type the entire host name, and after a few seconds `nslookup` will give you the address.

Once you have connected, you will need to go to the appropriate directory, lists its contents, and retrieve the files.

Most of the machines listed here run Unix, and you use "ls" and "cd" to list files and to change directories. Ymir runs VMS, and you will have to put square brackets around directory names, like [this].

Remember that although METAFONT sources are text files, pk fonts are not ASCII, and you will have to use binary mode for them. In general, use text mode for RE-ADME files and *.mf files, and binary mode for other font files. Files ending in .Z are compressed binary files – you will need to use binary mode, and then uncompress the files when you get them.

You can get files from ymir by sending mail messages to `mailserv@ymir.claremont.edu`

For example,

```
send [tex.mf.misc]cmapl10.mf
```

will get the file `cmapl10.mf` from the directory "tex.mf.misc". You can only get text files in this way.

7 Fonts

7.1 AMS (see under Euler)

The American Mathematical Society has adopted \TeX , and has had some fonts designed especially by Herman Zapf. These are known as the Euler fonts, and are described below. The METAFONT sources are now available by ftp.

NOTE that you should build these using `virnmf`, or at least with a version of METAFONT that does not have `cmbase` pre-loaded. You may also get errors reported when building some of the smaller sizes – simply press return.

There are

1. AMS Euler – a calligraphic font. This is not suitable as it stands for text use in \TeX , because it has a non-standard encoding.
2. AMS extra maths symbols
3. AMS computer modern extensions
4. AMS Cyrillic (this is the same as University of Washington Cyrillic)

There is also some documentation.

You can get them from the `ams` or from `ymir`: `ftp: e-math.ams.com [130.44.1.100] /ams/amsfonts/sources`

```
ftp: ymir.claremont.edu (134.173.4.23)
```

7.2 APL (A Programming Language)

```
ftp: ymir.claremont.edu (134.173.4.23):
```

```
cd SOFTWARE:[anonymous.tex.misc]
```

```
ftp: (the file is cmapl10.mf). Associated macros and documentation
```

```
ftp: are in SOFTWARE:[anonymous:tex:periodicals.tugboat] and also
```

```
ftp: [anonymous.tex.inputs.plain-contrib] in files apldef.tex,
```

```
ftp: aplstyle.tex, and aplverb.tex.
```

```
ftp: power.eee.ndsu.nodak.edu (134.129.123.1)
```

```
apl-tex-font/27-Jul-90
```

7.3 Babel – language support

The Babel archive at `ymir.claremont.edu` contains:

- **Greek Fonts:**

- Brian Hamilton Kelly's `cmgr` family

- Sylvio Levy's `gr` family [modern and classical]

- Yannis Haramboulos' `rgr` family

- **Hebrew fonts:**

- REDIS (a thin-looking sans serif)

- burkis (coming in January??)

- **Icelandic**

- Icelandic Modern (this is Computer Modern with extensions)

- **Russian (Cyrillic)**

- IFVE's `cmc` family

- U Washington's `wncyr` family

- Bashkiren/Mongolian (not sure if this is there yet)

- **Turkish**

- Turkish Modern (this is Computer Modern with extensions)

```
ftp: ymir.claremont.edu (134.173.4.23):
```

```
cd [anonymous.tex.babel]
```

7.4 Chess

Some chess typesetting macros were posted to `comp.text.tex` on July 31, 1990. The chess fonts are available for ftp from `ymir`.

```
ftp: ymir.claremont.edu (134.173.4.23):
```

```
cd [anonymous.tex.misc]
```

7.5 Chinese

$\text{Ch}\TeX$ is a set of macros and fonts for typesetting Chinese under \TeX . It only supports PostScript, and requires a modified `dvi2ps`. You write your article with PinYin.

```
ftp: 192.12.216.114 (name: chiris.stevens-tech.edu)
```

```
ftp: note that the usercode is "ftp", not "anonymous".
```

Included are source, fonts, VMS binary for 5.2, DOS binary, etc.

7.6 Bar Code

Dimitri Vulis's barcode font

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf]

7.7 Committee

This was produced during a workshop tutorial given by Donald Knuth. This is said (by Don Hosek, whom I trust in such things) to be hard-wired for the APS typesetter... It is described in Tugboat Vol. 5 No. 2 (Nov. 84).

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf.misc]
ftp: files are font1.mf and font1base.mf

7.8 Computer Modern

Computer Modern is Donald Knuth's font family used for his later 'Art of Computer Programming' books. It contains

cmr – computer modern roman
cmmi – computer modern maths italic
cmti – computer modern text italic
cmb – computer modern bold
cmss – computer modern sans serif
cmtt – computer modern typewriter
cmvtt – computer modern variable-spaced
typewriter

and several variations on each of the above (e.g. bold extended...). There are also some experimental fonts, such as cmff, a 'funny' font, and cmfib, a font based on Fibonacci numbers.

These are all included in a standard \TeX distribution, although I only know of one site archiving the META-FONT files separately. The official \TeX distribution site is labrea.stanford.edu:

ftp: labrea.stanford.edu [36.8.0.47]
cd pub/tex/cm
ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf.standard]

Many \TeX ftp archive sites also give access to the bitmaps (pk files), although you usually have to get them all at once.

There is also a reparameterised version of CM by John Sauter, which makes it easy to generate fonts at desired sizes— this is especially useful for non- \TeX users.

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf.sauter] See also: Babel, Sauter

There are some more Computer Modern variants stored at ymir; these include parameters for generating cm fonts at sizes 14, 18, 24 and 36pt. ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf.variants]

There is also a Pica typewriter font – italic and bold come out with straight and wavy underlines (respec-

tively), however. This is in ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf.pica]

Versions of these fonts suitable for use with an X Windows previewer (xtex and See \TeX) are available from ftp: foobar.colorado.edu pub/SeeTeX

7.9 Concrete

This font was designed for Donald Knuth's Concrete Mathematics book. It looks a little like a cross between American Typewriter and Computer Modern Roman. There are Roman and Italic faces.

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf]

7.10 Cyrillic

There are several Cyrillic (Russian-Alphabet) fonts:

- Tom Ridgeway's Cyrillic fonts
[BITNET: LISTSERV@UBVM.CC.BUFFALO.EDU]
- WN-Cyrillic
ftp: june.cs.washington.edu (128.95.1.4)

There is a mailing list, rustex-l, for discussion of typesetting Cyrillic-based languages. To subscribe, send mail to listserv@ubvm.bitnet containing the text SUBSCRIBE RUSTEX-L <your name here> or send mail to

Dimitri Vulis,
DLV%CUNYVMS1.BITNET@cunyvm.cuny.edu

7.11 Duerer

Based on the 16th Century drawings of Albrecht Duerer – see, for example, his "The Painter's Manual" published in Fac Simile by Abaris Books. Upper case only.

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf.duerer]

7.12 Devanagari

(this is a font for use with Sanskrit)

Created by Frans Velthuis in 1987/88 and is available from him

(Velthuis%hgrug5.earn@cunyvm.cunyvm.edu) for a small charge. Another address is "velthuis@hgrug5.bitnet".

F.J. Velthuis,
Nyensteinheerd 267
9736 TV Groningen
The Netherlands

7.13 Dingbats

By Doug Henderson.

ftp: ymir.claremont.edu (134.173.4.23):
cd [anonymous.tex.mf]

7.14 Euler

These were designed by Herman Zapf for the American Mathematical Society. See AMS.

The fonts are Fraktur, Script, Upright Italic, Math extension. (see TUGboat Vol. 10 No. 1)

7.15 Greek

Sylvio Levy's Greek font.

ftp: xydeco.siemens.com (129.73.1.101) – greek*

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.babel.greek.levy] (But Don Hosek tells me this is really Brian Hamilton Kelly's Greek font)

Also the Sylvio Levy 'gr' greek family, for both Modern and Classical Greek. This includes roman, typewriter and bold, plus \TeX macros.

Brian Hamilton Kelly's cmgr family

Yannis Haramboulos' rgr family

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.babel.greek]

See also Babel.

7.16 Hebrew

There is a sans-serif Hebrew font called REDIS available from ymir. It comes in three faces, including italic (slanting to the right, of course).

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf]

7.17 Helvetica

Produced by the Metafoundry and sold commercially. Unfortunately they do not distribute the METAFONT sources, so they are not really very useful in this file ...

The Metafoundry,

OCLC Inc., MC 485
6565 Frantz Road
Dublin, OH 43017
USA
+1 614 764-6087

7.18 Hershey

The Hershey fonts were designed for use by plotters, and published in 1972. These fonts are of relatively low typographic quality, but are useful on devices with lower resolution.

ftp: cs.uoregon.edu [128.223.4.13]

(but these are not in METAFONT – does anyone know where to get the METAFONT versions?)

7.19 International Phonetic Alphabet

Used by linguists, and also in some dictionaries. It's designed to go with Computer Modern. From WSU.

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf]

7.20 Music

Mu \TeX is a basic music package for \TeX . It seems to be the same as mtex, but with documentation in English rather than (or as well as) German.

ftp: stolaf.edu [130.71.128.1]

/pub/MuTeX.tar.Z /pub/MuTeX_doc.Z

ftp: suned.zoo.cs.yale.edu [128.36.21]

ftp: cs.ubc.edu [128.189.97.5]

src/MuTeX/MuTeX.tar.Z (slow)

ftp: wuarchive.wustl.edu: /mirrors/msdos/tex/mutex.arc

ftp: ymir.claremont.edu [134.173.4.23]:

cd [anonymous.tex.musix.mutex]

In Europe,

ftp: sol.cs.ruu.nl [131.211.80.5] in subdirectory pub/TEX

ftp: mtex.tar.Z (sources, including METAFONT sources and documentation)

ftp: mtexfonts.tar.Z (300dpi pk files) These are also available in Europe by mail-server. Send mail to mail-server@cs.ruu.nl with HELP in the subject and the body and probably the .signature as well :-)

7.21 OCR (Optical Character Recognition)

OCR A (this is not the font for printing on cheques)

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf]

7.22 Oriya

Sabita Panigrahi is working on Oriya (one of the modern Indian scripts), but this is not yet available.

7.23 Pandora

This is a little like Palatino, and includes a sans-serif variant. It is not yet of production quality, unfortunately.

ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf]

7.24 Pica

There is also a Pica typewriter font – italic and bold come out with straight and wavy underlines (respectively), however. This is in ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf.pica]

7.25 Pointing Hands

Various manual extremities, designed by Georgia Tobin
 ftp: ymir.claremont.edu (134.173.4.23):
 cd [anonymous.tex.mf]

7.26 Punk

A punK hAndWritten fOnT..
 For writing on walls, perhaps. The lower case is simply
 a smaller version of the upper case.
 ftp: ymir.claremont.edu (134.173.4.23):
 cd [anonymous.tex.mf]

7.27 Sanskrit

(see Fonts: Devanagari)

7.28 Sauter

This is a rework of Computer Modern. The outlines are
 the same, as far as I know, but it is much easier to gene-
 rate the various fonts.
 By John Sauter.
 ftp: ymir.claremont.edu (134.173.4.23):
 cd [anonymous.tex.mf]

7.29 Tengwar

There are at least two Tengwar fonts, Mike Urban's:
 ftp: ymir.claremont.edu (134.173.4.23):

cd [anonymous.tex.mf.tengwar]

Julian Bradfield <jcb@lfcs.edinburgh.ac.uk>
 also has both Tengwar and a prototype version of Cirth
 runes. Email only, although I have a copy and can mail
 them to you if you can't reach Julian. I prefer Juli-
 an's Tengwar, I think, but I am not an expert on Tolkien
 fonts...

7.30 Thai

There are at least two Thai fonts around. The rmit
 font is a little tricky to make, as it needs to be in a
 directory called "thai2", because it refers to files called
 "../thai2/name".

ftp: ymir.claremont.edu (134.173.4.23):
 cd [anonymous.tex.babel.thai.rmit]
 ftp: ymir.claremont.edu (134.173.4.23):
 cd [anonymous.tex.babel.thai.usl]

7.31 Vietnamese

ftp: blackbox.hacc.washington.edu [128.95.200.1]
 cd /pub/testviet

This test package includes tfm and pk fonts at 10 point
 roman and italic. I understand that you need T_EX 3.0 or
 later to use this, and drivers that cope with fonts contain-
 ing more than 128 characters.

BIJLAGE V**Program text generation with T_EX/L^AT_EX¹****Piet van Oostrum**

piet@cs.ruu.nl

March, 1991

Inhoud:

1. Web
 - 1.1 Cweb
 - 1.2 Fweb
 - 1.3 Spiderweb
2. Tgrind
3. C2latex
4. C++2latex
5. Cprog/Csty macros
6. Program environment
7. Schemetex
8. Ada
9. Miscellaneous

1 Web**1.1 Cweb****1.1.1 ctangle, cweave - translate CWEB to C and/or T_EX**

The ctangle program converts a CWEB source document into a C program that may be compiled in the usual way. The output file includes #line specifications so that debugging can be done in terms of the CWEB source file.

The cweave program converts the same CWEB file into a T_EX file that may be formatted and printed in the usual way. It takes appropriate care of typographic details like page layout and the use of indentation, italics, bold-face, etc., and it supplies extensive cross-index information that it gathers automatically.

CWEB allows you to prepare a single document containing all the information that is needed both to produce a compilable C program and to produce a well-formatted document describing the program in as much detail as the writer may desire. The user of CWEB ought to be familiar with T_EX as well as C.

Don Knuth wrote WEB for T_EX and Pascal. Silvio Levy designed and developed CWEB by adapting the WEB conventions to C and by recoding everything in CWEB.

1.1.2 Levy's Cweb system ported to MS-DOS and VAX/VMS

On the UK T_EX Archive at Aston University, directory [tex-archive.web.cweb] contains the sources for building Silvio Levy's CWEB system (a WEB suite which uses C as the programming language, and T_EX as the typesetting language) under Unix; this includes a bootstrapping version of Ctangle which permits one to get Cweave, etc., working.

Originally this was accompanied by a file VMS.CH which purported to port Cweb onto VAX/VMS: however, Vax-C has been revised since Levy released this, and further changes were necessary to get the system working under VMS. The original VMS.CH has therefore been deleted, and a new subdirectory [tex-archive.web.cweb.vms] added to hold the files for

¹Deze NTG bijlage beschrijft diverse methoden om programma-tekst in een T_EX/L^AT_EX document op te nemen. Voor het grootste gedeelte is deze beschrijving overgenomen uit de documentatie van de betreffende programma's resp. macro-pakketten. De simpelste manier is natuurlijk het gebruik van 'verbatim' en soortgelijke faciliteiten. Het nadeel van deze methode (en misschien tegelijk ook weer een voordeel) is dat er slechts gebruik gemaakt wordt van een vast font zonder variabele spatiering. De hierna genoemde methoden proberen de programmatekst te formatteren op een wat meer flexibele manier.

bootstrapping CWEB onto VMS, and the change files to make this Ctangle and Cweave.

In addition, I have ported CWEB to MS-DOS (under Borland's Turbo-C V1.5). This was a *major* undertaking, because of clashes between identifiers used in ANSI-standard function prototypes in Borland's libraries and constants defined in the .web files. Other very extensive changes were required to handle the large data structures, which exceed the 64kB segment limit of the PC architecture. However, it's all working eventually, and the files required for bootstrapping it, and the change files, will be found in [tex-archive.web.cweb.ms-dos].

For both of the above ports, the majority of the files of the Unix distribution will also be required.

To get started, fetch the file

```
[tex-archive.web.cweb]00readme.txt
from Aston, either by NIFTP (username PUBLIC,
password PUBLIC), or by sending mail to
<TeXserver@Uk.Ac.Aston.TeX>,
with the body of the message consisting of the two lines:
FILES
[tex-archive.web.cweb]00readme.txt
```

This file will tell you which other files will be required to be fetched to have a working version of CWEB under Unix, VMS or MS-DOS.

Brian Hamilton Kelly

1.2 Fweb, A Fortran/Ratfor/C version of WEB

Since the advent of Knuth's famous WEB system for documenting Pascal and, later, Levy's C version thereof, various Fortran users have inquired about the possibility of a WEB system for Fortran. I have developed a version of WEB that supports Fortran, Ratfor, and C (which can be mixed in the same WEB run). The new version is called FWEB. I am releasing v. 0.99 for beta-testing. This release is a bit premature (there are a few known bugs), but professional responsibilities force me to put FWEB on the back burner for a few months.

This project was driven by necessity. I was developing a large scientific code that I planned to write mostly in C. I was eager to document it as well as I could (so my graduate students could understand it), so I was interested in Silvio Levy's CWEB. Levy graciously gave me v. 0.5 of CWEB, which I used successfully for a few years.

However, part of my code could not be efficiently written in C; Fortran was necessary in some form. Straight Fortran is terrible, but R^ATional FORtran (Ratfor) removes many of the deficiencies and makes Fortran look something like C. Thus, part of the code was written in Ratfor, with a preliminary pass through the macro preprocessor m4 for good measure. Of course, I wanted to document that code as well. Now although CWEB was never intended for Ratfor, because of Ratfor's C-like syntax CWEB

was able in many cases to format Ratfor code satisfactorily. Of course, sometimes the documentation fell apart completely because keywords like DIMENSION aren't known to C, and certain syntax is different. Thus, after living with usable but somewhat marginal woven Ratfor output from CWEAVE for a while, I decided to do things right and teach CWEAVE the appropriate rules for Fortran. I thought that would take a couple of weeks...

The first project was to endow FWEB with the concept of a current language. Since I routinely mix C and Ratfor code and I wanted all the documentation to be all in one place, it was important to be able to switch between syntaxes as necessary. After some MONTHS (FWEB was developed in my spare time), I had taught the WEAVE processor to produce reasonable quality typeset Fortran and Ratfor, and to switch back and forth between the various languages. Little had been done to TANGLE at that point. However, I then decided that it was pointless to make separate passes through m4 and Ratfor; why not make TANGLE do it all. Levy had eliminated the macro preprocessor from CTANGLE since C has its own. I reinstated one, patterned after ANSI C. I also added a statement translator, so Ratfor keywords could be expanded.

Thus, the present version of FWEB has these significant enhancements over CWEB:

1. The concept of a current language (Fortran, Ratfor, and/or C);
2. A C-like macro preprocessor;
3. Ratfor to Fortran-77 translation.

My students and I use FWEB every day; we find it to be a tool of great utility. For writing in Fortran, the macro processor is indispensable. As far as Ratfor is concerned, I find it to be a great step forward; I hope never again to have to write a straight Fortran program.

The goal for FWEB v. 1.0 was to achieve functionality. It would be pointless to defend the elegance of all of the internal code at this point: some should be optimized for speed; some should be rewritten. Eventually, these projects will be undertaken. (Note that since I'm a major user I have a powerful motivation.) But now it's most important to get some Fortran users involved so I get some feedback and bug reports.

The relevant files are available via anonymous FTP from Internet host CCC.NMFECC.GOV in directory (VMS syntax) TEX\$ROOT:[DISTR.FWEB]. The files are described in READ.ME. As this explains, for VAX/VMS users only a subset of all files is necessary, since the executable binaries FTANGLE.EXE and FWEAVE.EXE are provided. If you have the courage to try to bring things up on another machine, you should also read INSTALL.FWEB, and transfer all files except *.EXE, *.HLB, and *.HLP to your machine.

If you are bootstrapping onto another machine, please note that you may have to make a few operating system-dependent changes in the source code. Feel free to con-

tact me for help.

I will appreciate bug reports, suggestions, etc. My addresses are

MFEnet: krommes@ppc.mfenet

Internet: Krommes%ppc.mfenet@ccc.nmfec.gov

Bitnet: krommes%ppc.mfenet@lbl.bitnet

John Krommes

1.3 Spiderweb

A note about a new implementation of WEB Norman Ramsey, Odyssey Research Associates, July 4, 1988²

1.3.1 abstract

Literate programming has received recent attention in the *Communications of the ACM* [Bentley 86, Van Wyk 87]. WEB is a tool intended for literate programming, but until recently it was useful only for writing PASCAL programs. The author has developed a new tool, SPIDER, which reads a description of a programming language and writes a WEB system that can be used to write programs in that language. SPIDER has been used in the author's organization to build WEB systems for Ada, C, AWK, and other languages. The author hopes that SPIDER will enable people to write literate programs in many more languages than they could before.

1.3.2 Introduction

Donald Knuth developed the WEB system of structured documentation as part of the \TeX project [Knuth 84]. His implementation of WEB combined PASCAL and \TeX . The WEB idea suggests a way of combining *any* programming language with *any* document formatting language, but until recently there was no software support for writing in WEB anything but PASCAL programs. In 1987, Silvio Levy rewrote the WEB system in C for C, while retaining \TeX as the formatting language [Levy 87]. I have modified Levy's implementation by removing the parts that make C the target programming language, and I have added a third tool, SPIDER, which complements WEAVE and TANGLE. SPIDER reads a description of a programming language, and writes source code for a WEAVE and TANGLE which support that language. Using SPIDER, a C compiler, and an AWK interpreter, an experienced systems programmer can generate in a few hours a WEB system for an Algol-like language.

²Copyright 1989 by Norman Ramsey, Odyssey Research Associates. To be used for research purposes only. For more information, see file COPYRIGHT in the parent directory.

1.3.3 Features of Spidery WEB

An exhaustive list of Spidery WEB's features would interest only WEB experts, but I do want to mention some features that I hope will encourage people to use Spidery WEB.

- TANGLE and WEAVE can read from multiple files (this feature is present in Levy's CWEB), and TANGLE can write to multiple files. Included files will be searched for on a path if not found in the current directory. These features make Spidery WEB more usable on systems that have make.
- TANGLE can expand macros with multiple parameters.
- The starred sections in Spidery WEB can be organized hierarchically (in three levels). We have a UNIX tool that can extract different pieces of the hierarchy from the output of WEAVE, so that it is possible to take excerpts from WEB documents.
- TANGLE writes `#line` directives, so you can debug at the WEB source level if your compiler respects the C conventions for `#line`.
- Many features of WEB seem to exist only to compensate for deficiencies in PASCAL, and most of those were dropped in CWEB. I have changed much of CWEB in order to avoid being bound too much by C conventions. As a result, there are dozens of minor differences between Spidery WEB and original WEB. To give just one example, Spidery WEB supports octal and hexadecimal constants using WEB-style notation, not the C notation used in CWEB.

1.3.4 Scope of SPIDER

SPIDER can generate WEB systems for a variety of languages. The author has written SPIDER description files for C, AWK, Ada, SSL (a language that describes attribute grammars to the Cornell Synthesizer Generator), the Larch Shared Language (a language for describing equational theories), and Dijkstra's language of guarded commands. Debugging the grammar that WEAVE uses to prettyprint the language is the most time-consuming part of creating a WEB system for a new target language, and SPIDER makes it trivial to change that grammar. To make a SPIDER description file for an Algol-like language that uses infix expression notation, an experienced systems programmer should be able to adapt an existing SPIDER description file very quickly.

SPIDER's major limitations are lexical. All Spidery WEBs assume that spaces and tabs in the input are not significant, except as separators; this makes it impossible to construct Spidery WEBs for languages like Fortran and Miranda, where the position of text on a line is significant. The lexical structures of identifiers, string literals, and numeric literals are fixed.

Conclusions

SPIDER is a modest piece of engineering; it does not introduce new ideas. SPIDER does make it possible to create a new WEB quickly, and to tinker with it easily. The author's group routinely uses Spidery WEB to write programs in Ada, C, and SSL, and has been pleased with the result. We have written in WEB an application of eighteen thousand lines, and we are very pleased at how easy it has been to review and maintain this code. The author hopes that the availability of Spidery WEB will encourage other groups to try literate programming, and that they, too, will be pleased with the results.

References

- [Bentley 86] Jon L. Bentley, "Programming Pearls," *Communications of the ACM* **29:5** (May 1986), 364–368, and **29:6** (June 1986), 471–483.
- [Knuth 84] Donald E. Knuth, "Literate Programming," *The Computer Journal* **27:2** (1984), 97–111.
- [Levy 87] Silvio Levy, "WEB Adapted to C, Another Approach," *TUGBoat* **8:1** (1987), 12–13.
- [Van Wyk 87] Christopher J. Van Wyk, "Literate Programming," *Communications of the ACM* **30:7** (July 1987), 593–599, and **30:12** (December 1987), 1000–1010.

2 Tgrind

Several people have asked about including program source in T_EX documents. I believe the most general approach is to use a utility like tgrind (TeX analog of vgrind). One specifies the language to be processed (so that tgrind can detect keywords). Tgrind converts tabs into the appropriate spacing (generates things like `\tab{24}` for 3 tabs), boldens keywords, prints quoted strings in typewriter font, prints comments in italics, and other nice things. All these goodies are customizable. This requires an extra pass, but the preprocessing is quite fast.

Tgrind is on the Unix T_EX tape, in the directory TeX-contrib/van. It was written by Van Jacobson of LBL. Unfortunately it is Unix specific, but I think equivalents for other OS are no problem. Heck, just take your favourite pretty-printer and generate T_EX instead of prettified output.

Tgrind formats program sources in a nice style using T_EX. Comments are placed in italics, keywords in bold face and strings in typewriter font. Source file line numbers appear in the right margin (every 10 lines). The start of a function is indicated by the function name in large type in the right margin.

In regular mode tgrind processes its input file(s) and passes them to T_EX for formatting and output.

In format mode (i.e., when the flag is used), tgrind processes its input file(s) and writes the result to standard output. This output can be saved for later editing, inclusion in a larger document, etc.

Currently known are PASCAL, RATFOR, Modula-2, MODEL, C, ISP, Yacc, Prolog, Icon, T_EX, CSH, and Bourne Shell.

Author of Tgrind is Van Jacobson, Lawrence Berkeley Laboratory (based on "vgrind" by Dave Presotto & William Joy of UC Berkeley).

Ken Yap

<ken@rochester.arpa>

3 C2latex

C2latex provides simple support for literate programming in C. Given a C source file in which the comments have been written in L^AT_EX, c2latex converts the C source file into a L^AT_EX source file. It can be used to produce typeset listings of C programs and/or documentation associated with the program.

The C source given to c2latex usually has the following form. It starts with a large comment containing L^AT_EX commands that start a document along with any initial text. Then there is a sequence of comment and code pairs, with the comment explaining the code to follow. The source file is ended by a comment containing L^AT_EX commands that finish the document.

C2latex produces L^AT_EX source by implementing a small number of rules. A C comment that starts at the beginning of a line is copied unmodified into the L^AT_EX source file. Otherwise, non-blank lines are surrounded by a pair of formatting commands (`\begin{flushleft}` and `\end{flushleft}`), and the lines are separated by `*`. Each non-blank line is formatted using L^AT_EX's `\verb` command, except comments within the line are formatted in an `\mbox`.

The c2latex program is written in ANSI C and can be processed by c2latex to produce L^AT_EX source containing a typeset listing of itself. It has a copyright similar to those distributed with GNU software. c2latex is available from me as a shar file via electronic mail. If there is enough interest, I will request that the sources be placed on a public server.

John D. Ramsdell

ramsdell@celebes.mitre.org

4 C++2latex

The program c++2latex converts ANSI-C/C++ programs into L^AT_EX source.

It requires flex which can be found on various ftp sites, e.g. prep.ai.mit.edu. For those without flex and without the possibility to get one, I can email the flex'ed program.

Please notice that this program is under GNU Copyleft.

Description

c++2latex is a tool for generating L^AT_EX source from ANSI-C or C++ programs. It recognizes all keywords, strings, and comments. These recognized parts can be set in different fonts. c++2latex can generate complete L^AT_EX files which can directly passed through L^AT_EX or parts of L^AT_EX files which can be included in other files (either direct or by the `\input` or `\include` commands). The output filename is searched in various steps. First, if the `{-o, +output}` flag is given, the output is written to the file pointed to by the value of this flag. If the `{-t, +pipe}` option is given, the output is written to stdout. (It is an error to specify both options together.) If none of this options but an input pathname is given, the output is written to a file whose name is the last component of the input pathname with the substituted or else added suffix '.tex'. If the input is read from stdin and none of the above options is given, the output is written to '`<program-name>.tex`' with `<program-name>` being the name of this program.

5 Cprog/Csty macros

The cprog macros allow programs in C, C++, Pascal, and Modula-2 to be included directly into T_EX documents. Program text is set in a Roman font, comments in slanted, and strings in typewriter. Operators such as `<=` are optionally combined into single symbols like `\le`. Keywords are *not* emphasised—I find this ugly and distracting. (By purest coincidence it would also be very hard to do.)

These macros can be `\input` in plain T_EX or used as a style file in L^AT_EX. They provide a convenient alternative to `tgrind`, particularly for program fragments embedded in documents. Full instructions for use appear in the macro package itself.

This allows C programs to be formatted directly by T_EX. It can be invoked by `\cprogfile{filename}` or (in L^AT_EX) `\begin{cprog} ... \end{cprog}` or (in plain T_EX) `\cprog ... \end{cprog}`. In L^AT_EX, the alternative form `\begin{cprog*}` is allowed, where spaces in C strings are printed using the 'square u' character (like L^AT_EX `verbatim*`). In plain T_EX, you have to use `\csname cprog*\endcsname` for this (sorry). If you are using `\cprogfile`, say `\cprogttspacetrue` beforehand if you want this effect.

The formatting is (necessarily) simple. C text is set in a normal Roman font, comments in a slanted font, and strings in a typewriter font, with spaces optionally

made visible as the 'square u' symbol. Tabs are expanded to four spaces (this does not look good when comments are aligned to the right of program text). Some pairs of input characters appear as single output characters: `<< <= >> >= != ->` are respectively T_EX's `\ll \le \gg \ge \ne \rightarrow`. Say `\cprogpairsfalseto` to disable this.

You can escape to T_EX within cprog text by defining an escape character. The character `@` is suitable for C and Pascal. I have not tested other characters so they may interact badly with their existing definitions here. To define `@` as the escape character, do `\cprogescape@`. Then within text you can do `@` followed by T_EX commands. These commands will be in a TeX group with the `\catcodes` of `\{\}%` as normal. The commands are terminated by a newline, which is not considered part of the program text.

The fonts below can be changed to alter the setting of the various parts of the program. The `\cprogbaselineskip` parameter can be altered to change the line spacing. L^AT_EX's `\baselinestretch` is taken into account too. The indentation applied to the whole program is `\cprogindent`, initially 0. Before and after the program there are skips of `\beforecprogskip` and `\aftercprogskip`; the default values are `\parskip` and 0 respectively (since there will often be a `\parskip` after the program anyway).

If the source text is Pascal or Modula-2, say `\pascaltrue` or `\modulatrue` (respectively) before formatting it. This makes `(* *)` be recognised for comments instead of `/* */`. Braces `{}` are also recognised for Pascal. `\pascalfalse` or `\modulafalse` as appropriate restores the default of C.

This package works by making a large number of characters active. Since even spaces are active, it is possible to examine the next character in a macro by making it a parameter, rather than using `\futurelet` as one would normally do. This is more convenient, but the coding does mean that if the next character itself wants to examine a character it may look at a token from the macro rather than the input text. I think that all cases that occur in practice have been looked after.

The macros could still do with some work. For example, the big macro defined with `[]` taking the place of `{}` could be recoded to use `{}` and so be more legible. The internal macros etc should have `@` in their names, and should be checked against L^AT_EX macros for clashes.

Eamonn McManus

`<emcmanus@cs.tcd.ie>`

6 Program environment

In T_EX or L^AT_EX it is possible, but difficult, to create a nice layout for programs. The easiest way is to use the `verbatim` environment. The layout is then copied from the input file. In most books a program (e.g., a PASCAL-program) is displayed using boldface reserved words, using math-italic for statements and using teletype fonts for string representations. Without extra equipment this can be done in T_EX as well as in L^AT_EX using a `tabbing`-environment. Of course, each font choice should be made explicitly, e.g., you must say `{\bf begin}` to create a boldface begin-symbol and `$x:=x+1$` to denote an assignment. Furthermore, the user is responsible for setting the tabs and jumping to the right ones. This is far from being user-friendly.

The `program`-environment tries to be of some help while displaying program-texts. It contains a number of macros of the form `\BEGIN`, `\PROCEDURE` and alike that not only put down a boldface begin- or procedure-symbol but also sets and jumps to the right tabs. Using the `program` environment from the `program2`-style file also automatically puts the statements in math-mode (you do not have to use `$`-signs anymore).

Rein Smedinga

Department of computing science,
P.O. box 800,
9700 AV Groningen
rein@cs.rug.nl
November 9, 1990

7 Schemetex

SchemeTeX provides simple support for literate programming in any dialect of Lisp on Unix. Originally created for use with Scheme, it defines a new source file format which may be used to produce L^AT_EX code or Lisp code. Roughly speaking, L^AT_EX formats Lisp code in a verbatim-like environment, and it formats Lisp comments in an ordinary environment.

SchemeTeX is available via anonymous FTP from `linus (192.12.120.51)` in the `shar` file named `"pub/schemeTeX.sh"`. Included is an operating system independent version for the T dialect of Lisp.

John D. Ramsdell

`<ramsdell%linus@mitre-bedford.ARPA>`

8 Ada

Zie TUGboat 10#1, April 1989

APE – A set of T_EX macros to format Ada programs

I have developed a set of macros to do exactly this for Ada programs. To get more details on these macros, read the report in the April issue of TUGBoat, or the Nov/Dec issue of Ada Letters (both in 1989).

The macros are available by anonymous ftp to `anna.stanford.edu` in the `pub` directory (I think). I can send them to you if you are willing to cover the costs.

Sriram Sankar

Zie ook:

GUTenberg'90

May 15-18, 1990

University Paul Sabatier, Toulouse, FRANCE

Typesetting ADA programs (P. Naudin, C. Quitte)

9 Miscellaneous

9.1 Typesetting programming languages in L^AT_EX

About two years ago I wrote a program that converts programs to T_EX. The program sets keywords in boldface (or any font you select) using a data file to find out which strings are keywords and how to skip comments. I have data files for Modula-2, C, C++, Pascal, Occam, Beta, and some more. The program is written in C (under VMS) and could well be improved. However it does what I want it to do, it makes programs 'look' nice in listing. As said it translates to T_EX and not L^AT_EX but that is no big deal to change I'd guess.

If you need this program send a mail to `rosenber@ra.abo.fi`. mail

Robin Rosenberg

9.2 Typesetting PASCAL in L^AT_EX

I have an SED script and Pascal environment for L^AT_EX that follows the standard Algol 60 style for setting Pascal text. The SED script translates everything between `\begin{pascal}` and `\end{pascal}` in various ways that the pascal environment understands. The only problem with the thing is that indenting must be in multiples of 4 spaces, but I've used it for a number of publications without hearing any complaints about my awkward indenting style.

In any case, I strongly recommend the notion of environments for language types, as opposed to the various grind programs. It would be nice to have a standard set of environment parameters to control things like keyword font so language environments from different sources could be at least somewhat interchangeable.

Doug Jones

`jones@herky.cs.uiowa.edu`

9.3 manpage.sty

This style option is designed to work with the report document style of L^AT_EX version 2.09. Use `\documentstyle[11pt,manpage]{report}`

```
\begin{manpage}{Title}{Module}{Version} % see an example, all will be clear
\end{manpage} % end of manpage environment
\variable{#1} (e.g., \variable{int foo}) % with \medskip added
\variable*{#1} (e.g., \variable*{int bar})% no extra spacing
\function{#1} (e.g., \function{void demo(int dummy)}) % with \medskip added
\function*{#1} (e.g., \function*{void demo(int dummy)})% no extra spacing
\subtitle{#1} (e.g., \subtitle{AUTHOR}) % fit in the same line if possible
\subtitle*{#1} (e.g., \subtitle*{AUTHOR})% always break a newline
"#1" (e.g., "dummy_variable") % argument is in italic&unbreakable
\separator % draw a thin line to separate subtitle from the text
\header{#1}{#2}{#3} % in case you want to have a header and
\footer{#1}{#2}{#3} % a footer outside of the manpage environment
\dq % print double quote character (")
```

This L^AT_EX style file is similar to the UNIX troff man macros in format and is specially tuned for documenting the C++ library that the author wrote.

The commands that are created in the style file are:

In the `\function` macro, data types and their dummy arguments are separated by a space. So if you have a function like "int f(const int n)", you should document it as `\function{int f(const~int n)}`. The argument n is optional. In the `\subtitle` macro, two lines of text may be divided by "\\".

Rong Chen (rchen@cs.uiuc.edu)
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801

9.4 Typesetting Scheme code

I wrote something called S^LA^TE^X that allows listings of Scheme code in L^AT_EX without restricting it to the usual monospace typewriter font supported by other code typesetters.

I've placed the current version in the titan.rice.edu's anonymous ftp area: `get public/slatex.sh`.

S^LA^TE^X decides which tokens should be, say, boldfaced, italicized, or sansserifed, pretty much along the style of the Little Lisper [1]. (The user can completely control this default decision process, so much so that he can flip the fonts around, add new fonts, or even do something silly like make everything come out in typewriter – i.e., turn the program into a no-op.)

I'd been leery of distributing S^LA^TE^X before because of the frequent updating that it's undergone following the Rice Scheme-and-similar- language users' lively demands for bells and whistles, e.g.,

- allowing arbitrarily positionable displays, boxed code, in-text code, and directly inputting actual Scheme files;
- getting little pockets of L^AT_EX text or mathmode into the Scheme code, for readable Schemelike pseudo-code (useful for expository papers and class hand-outs);
- making it learn automatically that a macro definition implies that keyword should henceforth be boldfaced, etc.

At any time, the ftp site will contain the most recent code. The shar file contains the Scheme source (Ch*z, but should carry over to other Schemes with minor changes), a shellscrip that piggybacks the codesetter on to L^AT_EX, the requisite L^AT_EX style file, installation instructions, a manual in L^AT_EX, a man page, and a copyleft. The first 2 sections in the manual suffice for most uses, with fine tuning being described later. (The code, as of now, contains more fine tuning than documented – I'll update the docs when I next get time.)

This is free(ly distributable) software, and hence no warranty, though I'll be glad to field bug and other reports.

[1] D.P. Friedman and M. Felleisen, 'The Little Lisper', Science Research Associates (3e), 1989, and The MIT Press, 1987.

Dorai Sitaram
dorai@tone.rice.edu

BIJLAGE W

L^AT_EX for engineers and scientists

(book review)¹

Nico Poppelier

n.poppelier@elsevier.nl

Februari, 1991

Although the L^AT_EX manual is a useful book, it is not suitable as an introduction, as a book for beginning users. ‘L^AT_EX for engineers and scientists’ by David J. Buerger, which was published this year, at first sight appears to be a good introduction. In the preface the author writes: ‘[this book] was written to provide a fast and easy way to learn how to produce technical documents with L^AT_EX.’ And indeed, ‘L^AT_EX for engineers and scientists’ is a book that doesn’t frighten readers by its length and is easy to read. It describes B_IB_TE_X and M_Ak_EI_Nd_Ex, it gives exercises – with answers – that are really not bad, and it contains an index – although I find that a bit short – and a glossary.

Unfortunately my general opinion about this book is not positive: both the contents of the book and the quality of the book as a printed product leave a lot to be desired.

My overall impression of the contents of the book can be summarized in a few points.

- The author has not quite grasped the concept of a document style and the separation between logical and visual structure, two fundamental concepts of L^AT_EX.
- The author does not distinguish between L^AT_EX proper and L^AT_EX *plus* the standard document styles. There are many document styles beside the standard ones, so this distinction is essential.
- In several examples L^AT_EX and T_EX commands are mixed. My opinion is that in examples only L^AT_EX commands should be used. If the author insists on mentioning the T_EX equivalents, he should explain what sort of functionality L^AT_EX adds.
- Some functions of L^AT_EX, among which at least one important function, are not explained in the book.
- Explanations in the book are sometimes confusing or sloppy. In a few cases they are even incorrect.

I will give some examples:

- In chapter 4, Formatting environments, the author starts with the `center`, `flushleft` and `flushright` environments, and then goes on to treat the list and quotation environments. The main

purpose of L^AT_EX’s markup instructions is describing the logical structure of the text. In a book on L^AT_EX, descriptions of logical design should come before descriptions of visual design.

- The custom description list on p. 28 refers to layout parameters of the `list` environment that cannot be found in the index or the glossary. Although the author includes in his book instructive page-layout diagrams² that are unfortunately absent in the L^AT_EX manual, he forgets to include the equally useful list-layout diagram that is printed on p. 113 of the L^AT_EX manual.
- The custom description list given as an example on p. 28 is a variation on the `description` environment described in the L^AT_EX manual. In this example the items are typed as `\item[{\bf Fox}]`; as a result there is no clear separation between form and contents. A better way would be to define the layout of the items in the definition of the customised list. That way, one only has to type `\item[Fox]`.
- On p. 39 the author gives a table of the typeface sizes that correspond to L^AT_EX commands such as `\small`, `\normalsize` and `\large`. The correspondence given in the table is valid only for the standard document styles and not for *every* document style. By failing to make this distinction, the author suggests that the table is universally valid, which it isn’t.
- In chapter 6 the author treats only the `$... $` and not the `\(... \)` construction for in-line mathematical formulae. `$... $` and `$$... $$` give formulae in a more or less fixed layout. If one uses L^AT_EX’s `\(... \)` and `equation` environment instead, the user lets the document style control the formula layout. Furthermore, the L^AT_EX notation for formulae has opening and closing tags that are not identical, which results in fewer errors.
- In chapter 7, on p. 52, the author introduces the `\lefteqn` command without any explanation. This is a command that a lot of users find confusing: they often think that `\lefteqn` puts an equation flush with the left margin of the text.

¹To be submitted to TUGboat, © 1991, T_EX Users Group.

²Similar diagrams have appeared in TUGboat.

- In chapter 8 the author gives a confusing description of the two environments `table` and `tabular`. The `tabular` environment produces a table, i.e. an arrangement of cells in rows and columns, possibly with horizontal and vertical rules³. The `table` environment creates a floating object, i.e. a part of the document for which L^AT_EX tries to find a good place to print it. In most cases, the `table` environment contains a caption that starts with the word ‘Table’⁴ and a `tabular` environment for the actual table contents. However, Buerger writes (italics mine):

Tables *created* with the `tabbing` or `tabular` environments—...
The `\begin{table[]}` or `\begin{figure[]}` command will *create* a table or figure.

- On p. 64 the author explains the use of `\label` and `\ref`. He instructs the reader to put the `\label` command after sectional-unit commands and after the `\caption` command of a `figure` or `table` environment. However, there is no information on where to put the label in `equation` and `eqnarray` environments.
- In chapter 10, Organizing a document, the author uses in an example


```
\topmargin 0mm
\def\BibTeX{ ... }
```

 instead of the L^AT_EX equivalents


```
\setlength{\topmargin}{0mm}
\newcommand{\BibTeX}{ ... }
```
- In chapter 10 the author fails to distinguish between L^AT_EX proper and the standard document styles. On p. 68 the author writes:

Title information is automatically centered.

and (italics by the author):

You can produce an abstract placed below the title information ... by typing the following command *before* the `\maketitle` command.

In both cases the behaviour the author describes is that of the standard document styles: in other document

styles a title could be left-justified and emphasized phrases could be printed in a boldface font. In the second case, the author is also definitely wrong since the `\maketitle` command defined in the standard document styles does not print the abstract, but only the title, author and date.

The author is also inconsistent with notation: for example, in pages vii-xiii, the table of contents, list of figures and list of tables, I found ‘L^AT_EX’, ‘LaTeX’ and ‘LaTeX’! I sometimes got the feeling that the book was written or at least finished in some haste.

Some examples of features of L^AT_EX that are missing in ‘L^AT_EX for engineers and scientists’.

- The author writes that the `\include` command is similar to the `\input` command, except that it starts on a clean page. He doesn’t mention one of the nicest mechanisms in L^AT_EX: cross-referencing between sub-documents if some of the sub-documents are excluded from the current formatting run by means of `\includeonly`.
- The only information on T_EX’s units was the sentence ‘There are 72.27 points to an inch’, and I found it in the chapter on error messages!
- One of the sample input files contains the `\;` command, without explanation and without treating other, similar commands.

So far, I have only criticized the author. However, I think the publisher of this book, McGraw–Hill, can be blamed for a few things as well. Concerning the quality of the book as a printed product: the book was produced from camera-ready pages prepared by the author on a laser printer. Computer Modern is a good typeface, if only you use it on a printing device of sufficiently high quality. Laser printer quality is, I’m afraid, not good enough and I hope this book is one of the last books on T_EX-related matters produced in such a way. As for the contents of the book: it seems likely that McGraw–Hill did not ask an expert to review the book, otherwise they would have asked the author to rewrite parts of it.

‘L^AT_EX for engineers and scientists’ is not a bad book, but it is not a good book either. It can be used, but I can’t really commend it.

³An imprecise definition of a table, I know!

⁴To be precise: this is specified by the document style, but it should be ‘Table’ or something equivalent.

BIJLAGE X**The structure of the \TeX processor**^{1 2}**Victor Eijkhout**

Center for Supercomputing Research and Development
 University of Illinois
 305 Talbot Laboratory
 104 South Wright Street
 Urbana, Illinois 61801-2932, USA
 eijkhout@csrd.uiuc.edu

Februari, 1991

The inner workings of \TeX are explained by its author [1] in terms of an analogy with the digestive tract. Apart from the fact that this gives rise to a whole genre of jokes³, the analogy becomes definitely strained when regurgitation takes place in the mouth, or when the eyes take part in the process.

In this article I will describe the \TeX processor as a multi-layered engine that successively transforms characters into tokens, tokens into lists, and from these lists builds a typeset page.

1 Four \TeX processors

The way \TeX processes its input can be viewed as happening on four levels. One might say that the \TeX processor is split into four separate units, each accepting the output of the previous stage, and delivering the input for the next stage. The input of the first stage is then the `tex` input file; the output of the last stage is a `dvi` file.

For many purposes it is most convenient and insightful to consider these four levels of processing as happening after one another, each one accepting the *completed* output of the previous level. In reality this is not true: \TeX is not something like a four-pass compiler. All levels are simultaneously active, and there is interaction between them.

The four levels are

1. The input processor. This is the piece of \TeX that accepts input lines from the file system of whatever computer \TeX runs on, and turns them into tokens. These are mostly character tokens that comprise the typeset text, and control sequence tokens that are commands to be processed by the next two levels.
2. The expansion processor. A number of tokens gene-

rated in the first level – macros, conditionals, and a number of primitive \TeX commands – is subject to expansion. Expansion is the process that replaces some (sequences of) tokens by another (possibly empty) sequence.

3. The execution processor. Control sequences that are not expandable are executable, and this execution takes place on the third level of the \TeX processor. One part of the activity here concerns changes to \TeX 's internal state: assignments and macro definitions are typical activities in this category. The other thing going on on this level is the construction of horizontal, vertical, and mathematical lists.
4. The visual processor. In the final level of processing the visual part of \TeX processing is performed. Here horizontal lists are broken into paragraphs, vertical lists are broken into pages, and formulas are built out of math lists. Also the output to the `dvi` file takes place on this level. The algorithms working here are not accessible to the user, but they can be influenced by a number of parameters.

2 The input processor

The input processor is that part of \TeX that translates whatever characters it gets from the input file into tokens. The output of this processor is a stream of tokens: a token list. Most tokens fall into two categories: character tokens and control sequence tokens. The remaining category is that of the parameter tokens; these will not be treated here.

¹To be published in TUGboat, © 1991, \TeX Users Group.

²This is a chapter from my book ' \TeX by Topic', to be published by Addison-Wesley.

³Tokens being 'sicked up again' [2], output being ' \TeX crement' [3], or the particularly deplorable title of [4] . . .

2.1 Character input

For simple input text, characters are made into character tokens. However, \TeX can ignore input characters: a row of spaces in the input is usually equivalent to just one space. Also, \TeX itself can insert tokens that do not correspond to any character in the input, for instance the space token at the end of an input line, or the `\par` token after an empty line.

Not all character tokens represent characters that are to be typeset. Characters fall into sixteen categories – each one specifying a certain function that a character can have – of which only two contain the characters that will be typeset. The other categories contain such characters as `{`, `}`, `&`, and `#`. A character token can be considered as a pair of numbers: the character code – usually the ASCII code – and the category code.

When the escape character `\` appears in the input, \TeX 's behaviour in forming tokens is more complicated. Basically, \TeX builds a control sequence by taking a number of characters from the input and lumping them together into a single token.

The behaviour with which \TeX 's input processor reacts to category codes can be described as finite-state automaton with three internal states: N , new line, M , middle of line, and S , skipping spaces. These states and the transitions between them are treated in chapter 8 of *The \TeX book*.

2.2 Two-level input processing

\TeX 's input processor is in fact even a two-level processor. Due to limitations of the terminal, the editor, or the operating system, the user may not be able to input any desired character. Therefore, \TeX provides a mechanism to access with two superscript characters all of the available character positions. This may be considered to be a separate stage of \TeX processing, taking place prior to the three-state finite automaton mentioned above.

For instance, the sequence `^^+` is replaced by `k` because the ASCII codes of `k` and `+` differ by 64. Since this replacement takes place before tokens are formed, one may write `\vs^^+ip 5cm` to get the effect of `\vskip 5cm`. More useful examples than this exist.

Note that this first stage is a transformation from characters to characters, without considering category codes. These come into play only in the second phase of input processing, where characters are converted to character tokens by coupling the category code to the character code.

3 The expansion processor

\TeX 's expansion processor accepts a stream of tokens and, if possible, expands the tokens in this stream one by one until only unexpandable tokens remain. Macro

expansion is the clearest example of this: if a control sequence is a macro name, it is replaced (together possibly with parameter tokens) by the definition text of the macro.

Input for the expansion processor is provided mainly by the input processor. The stream of tokens coming from the first stage of \TeX processing is subject to the expansion process, and the result is a stream of unexpandable tokens which is fed to the execution processor.

However, the expansion processor comes into play also when an `\edef` or `\write` is processed. The parameter token list of these commands is expanded as if the lists would have been on top level, instead of the argument to a command.

There is a special fascination to macros that work completely by the expansion processor. See the recent articles [4], [5], and [6] for some good examples.

3.1 The process of expansion

Expanding a token comprises the following steps:

- See if the token is expandable.
- If the token is unexpandable, pass it to the token list currently being built, and take on the next token.
- If the token is expandable, replace it by its expansion. For macros without parameters, and a few primitive commands such as `\jobname`, this is indeed a simple replacement. Usually, however, \TeX needs to absorb some argument tokens from the stream in order to be able to form the replacement of the current token. For instance, if the token was a macro with parameters, sufficiently many tokens need to be absorbed to form the arguments corresponding to these parameters.
- Go on expanding, starting with the first token of the expansion.

Deciding whether a token is expandable is usually a simple decision. Macros and active characters, conditionals, and a number of primitive \TeX commands (see the list on page 215 of *The \TeX book*) are expandable, other tokens are not. Thus the expansion processor replaces macros by their expansion, it evaluates conditionals and eliminates any irrelevant parts of these, but tokens such as `\vskip` and character tokens, including characters such as dollars and braces, are passed untouched.

3.2 Special cases: `\expandafter`, `\noexpand`, and `\the`

As stated above, after a token has been expanded \TeX will start expanding the resulting tokens. At first sight the `\expandafter` command would seem to be an exception to this rule, because it expands only one step. What actually happens is that the sequence

```
\expandafter<token1><token2>
```

is replaced by

```
<token1><expansion of token2>
```

and this replacement is in fact reexamined by the expansion processor.

Real exceptions do exist, however. If the current token is the `\noexpand` command, the next token is considered for the moment to be unexpandable: it is handled as if it were `\relax` (more about this control sequence follows below), and it is passed to the token list being built.

Example: in the macro definition

```
\edef\ a{\noexpand\ b}
```

the replacement text `\noexpand\ b` is expanded at definition time. The expansion of `\noexpand` is the next token, with a temporary meaning of `\relax`. Thus, when the expansion processor tackles the next token, the `\ b`, it will consider that to be unexpandable, and just pass it to the token list being built, which is the replacement text of the macro.

Another exception is that the tokens resulting from `\the<token variable>` are not expanded further if this statement occurs inside an `\edef` macro definition.

3.3 Braces in the expansion processor

Above, it was said that braces are passed as unexpandable character tokens. In general this is true. For instance, the `\romannumeral` command is handled by the expansion processor; when confronted with

```
\romannumeral1\ number\ count2 3{4
```

\TeX will expand until the brace is encountered: if `\count2` has the value of zero, the result will be the roman numeral representation of 103.

As another example,

```
\iftrue {\else }\fi
```

is handled by the expansion processor as if it were

```
\iftrue a\else b\fi
```

The result is a character token, be this a brace or a letter.

However, in the context of macro expansion the expansion processor will recognize braces. First of all, a balanced pair of braces marks off a group of tokens to be passed as one argument. If a macro has an argument

```
\def\macro#1{ ... }
```

one can call it with a single token

```
\macro 1 \macro \ $
```

or with a group

```
\macro {abc} \macro {d{ef}g}
```

Secondly, when the arguments for a macro with parameters are read, no expressions with unbalanced braces are accepted. In

```
\def\ a#1\ stop{ ... }
```

```
\ a bc{d\ stop}e\ stop
```

the argument is `bc {d\ stop} e`. Only balanced expressions are accepted here.

4 The execution processor

The execution processor builds lists: horizontal, vertical, and math lists. Corresponding to these lists, it works in horizontal, vertical, or math mode. Of these three modes ‘internal’ and ‘external’ variants exist. In addition to building lists, this part of the \TeX processor also performs mode-independent processing, such as assignments.

Coming out of the expansion processor is a stream of unexpandable tokens to be processed by the execution processor. From the point of view of the execution processor, this stream contains two types of tokens:

- Tokens that signal an assignment (this includes macro definitions), and other tokens that are independent of the mode, such as `\show` and `\aftergroup`.
- Tokens that build lists: characters, boxes, and glue.

Some objects can be used in any mode, for instance boxes can appear in horizontal, vertical, and math lists. The effect of such an object will of course still depend on the mode. Other objects are specific for one mode. For instance, characters (to be more precise: character tokens of categories 11 and 12) are intimately connected to horizontal mode: if the execution processor is in vertical mode when it encounters a character, it will switch to horizontal mode.

For the expansion processor a character token is just an unexpandable object. On this level, however, something is actually done with it. Some characters are typeset, but the execution processor can also encounter, for instance, math shift characters (usually $\$$), or braces. When a math shift character is found in the stream of tokens, math mode is entered (or exited if the current mode was math mode); when a left brace is found, a new level of grouping is entered.

One control sequence handled by the execution processor deserves special mention: `\relax`. This control sequence is not expandable, but the execution is ‘empty’. Compare the effect of `\relax` in

```
\count0=1\relax 2
```

with that of `\null` defined by

```
\def\null{}
```

in

```
\count0=1\null 2
```

In the first case the expansion process that is forming the number stops at `\relax` because it is unexpandable, and the number 1 is assigned. In the second case `\null` expands to nothing, so 12 is assigned.

5 The visual processor

\TeX 's visual processor encompasses those algorithms that are outside direct user control: paragraph breaking, alignment, page breaking, math typesetting, and dvi file generation. Various parameters control the operation of these parts of \TeX .

Some of these algorithms return their results in a form that can be handled by the execution processor. For instance, a paragraph that has been broken into lines is added to the main vertical list as a sequence of horizontal boxes with intermediate glue and penalties. Also, the page breaking algorithm stores its result in $\backslash\text{box}255$, so output routines can dissect it. On the other hand, a math formula can not be broken into pieces, and, of course, shipping a box to the dvi file is irreversible.

6 Further examples

6.1 Skipped spaces

Skipped spaces provide an illustration of the view that \TeX 's levels of processing accept the completed input of the previous level. Consider the commands

```
\def\ a{\penalty200}
\ a 0
```

Faulty reasoning

“The $\backslash\text{a}$ is encountered, expanded, the space then delimits the number”

would lead to the conclusion that this is equivalent to $\backslash\text{penalty}200 0$. It is not. Instead, what results is $\backslash\text{penalty}2000$

because the space after $\backslash\text{a}$ is skipped in the input processor.

6.2 Internal quantities and their representations

\TeX uses various sorts of internal quantities, such as integers and dimensions. These internal quantities have an external representation, which is a string of characters, such as 4711 or 91.44cm.

Conversions between the internal value and the external representation take place on two different levels, depending on the direction the conversion goes. A string of characters is converted to an internal value in assignments such as

```
\pageno=12 \baselineskip=13pt
```

or statements like

```
\vskip 5.71pt
```

and all of these statements are handled by the execution processor.

On the other hand, the conversion of the internal values into a representation as a string of characters is handled by the expansion processor. For instance,

```
\number\pageno \romannumeral\year
\the\baselineskip
```

are all processed by expansion.

Note that in the $\backslash\text{baselineskip}$ example above the conversion from string of characters to internal value was ‘automatic’. The conversion the other way has to be forced by a command such as $\backslash\text{number}$. Thus there is no danger that the sequence

```
\pageno=3 \count\MyCount=\pageno 5
```

will result in assigning either 15 or 35 to $\backslash\text{MyCount}$.

As a final example, suppose $\backslash\text{count}2=45$, and consider the statement

```
\count0=1\number\count2 3
```

The expansion processor tackles

$\backslash\text{number}\backslash\text{count}2$ to give the characters 45, and the space after the 2 is absorbed because it only serves as a delimiter of the number of the $\backslash\text{count}$ register. In the next stage of processing, the execution processor will then see the statement

```
\count0=1453
```

and execute this.

References

- [1] Donald Knuth, *The \TeX book*, Addison–Wesley Publishing Company, 1984.
- [2] Angela Barden, Some \TeX manuals, TUGboat 12(1991), no. 1.
- [3] Ron Whitney, private communication.
- [4] Victor Eijkhout, Oral \TeX , TUGboat 12(1991), no. 2.
- [5] Alan Jeffrey, Lists in \TeX 's mouth, TUGboat 11, no. 2, 237–245.
- [6] Sonja Maus, Looking ahead for a $\langle\text{box}\rangle$, TUGboat 11, no. 4, 613–614.

BIJLAGE Y**Comments on the Future of T_EX and METAFONT**¹**Nelson H.F. Beebe**

Center for Scientific Computing
 Department of Mathematics
 220 South Physics Building
 University of Utah
 Salt Lake City, UT 84112
 USA
 Tel: (801) 581-5254
 FAX: (801) 581-4148
 Beebe@science.utah.edu

October 1990

1 Introduction

Donald E. Knuth's article, "The Future of T_EX and METAFONT", elsewhere in this issue², clearly states the Grand Wizard's wishes about these programs and the Computer Modern font family.

Where does that leave TUG? The opening paragraph of TUG's bylaws includes this statement (the emphasis is mine):

...specifically to identify, develop, operate, fund, support, promote and encourage charitable, educational and scientific programs and projects which will stimulate those who have an interest in *systems for typesetting technical text and font design*; to exchange information of same and associated use of computer peripheral equipment; to establish channels to facilitate the exchange of macro packages, etc., through publications and otherwise; and to develop, implement and sponsor educational programs, seminars and conferences in connection with the foregoing. . .

I believe that this expressly says that TUG's purview legitimately goes beyond T_EX, METAFONT, and Computer Modern, whose further development has been frozen by their author in the interests of providing a constant solid base for their users, and of returning to his own extensive research and writing efforts, which have been outstanding landmarks in the development of the fields of Computer Science and Applied Mathematics.

2 T_EX is international

As the T_EX-related portion of the Utah bibliography project described in my President's message in this issue of *TUGboat* will attest, the use of T_EX is widespread. Many books and journals are routinely typeset by T_EX, including almost all of the publications of the American Mathematical Society, one of the world's largest publishers of mathematical material. Large on-line data bases in T_EX input form now exist.

I suggest that no other typesetting system, or desk-top publishing system, has been used for as many languages as T_EX has. T_EX is in use for all major European languages, plus Arabic, Chinese, Coptic (Ethiopian), Hebrew, several Indian languages, Japanese, Persian, Russian, Thai, Turkish, Vietnamese, and likely others that I may be unaware of. This list includes languages that are written horizontally and vertically. T_EX can support typesetting of multiple languages in the same text, thanks to the work of Frank Liang on hyphenation [11], of Michael Ferguson on multi-lingual TeX [4, 5, 6, 7], and of Donald Knuth and Pierre MacKay on T_EX- $\mathcal{X}\mathcal{E}\mathcal{L}$ [9].

These research efforts led to several features incorporated in T_EX 3.0 to make multilingual typesetting standardly available. For related work in other typesetting systems, see [2] on tri-directional typesetting, and articles in the July 1987, August 1988, and May 1990 issues of the Communications of the ACM.

There are textbooks about T_EX in at least Danish, Dutch, English, French, German, and Japanese, and I know of

¹ Reprinted from TUGboat 11 (1990), No. 4 – © 1990, T_EX Users Group; reprinted with permission.

² Appendix GG of NTG MAPS 90.2.

in-progress translations to Persian of the T_EXbook and the L^AT_EX User's Guide and Reference Manual.

There are TUG members in nearly 50 countries, and I'm sure there are T_EX users in many more. Besides TUG, there are five thriving regional groups in Western Europe, and five or more others are forming.

3 The challenge from desk-top publishing systems

The international use of T_EX suggests that Donald Knuth's decision to freeze further development will in some ways be highly beneficial. However, it does *not* imply that T_EX, METAFONT, and Computer Modern are the last word in computer-based typesetting. If TUG does not pursue further development of typesetting software, T_EX may be doomed to extinction far sooner than it should, for several reasons:

- Desk-top publishing is *big* business, with several tens of millions of installed personal computers forming the potential market base. The Salt Lake Tribune on 10 October 1990 carried an article on Utahns included in the just-released Forbes list of the 400 wealthiest people in the world. The two developers of Word Perfect, one of the most popular word processing systems available on personal computers, workstations, and some mainframes, have a combined worth of nearly one (North American) billion dollars; the young chairman of Microsoft Corporation is worth even more.
- Software is a commodity that is relatively cheap to produce and distribute. The actual development costs of most commercial software are only a small fraction of potential sales revenues, and the computing industry has numerous examples of the quick attainment of fabulous wealth. What *does* cost a lot of money is sales and marketing, and the on-going support of software, including personnel, authoring, and documentation. This situation encourages competitiveness and rapid development of new products.
- Desk-top publishing (WYSIWYG)³ systems are attractive to many people, particularly novices, because of the immediate feedback that they provide. With most of them, it is impossible to generate syntax errors of the type that T_EX is perhaps infamous for, because input is checked character by character as it is entered, and formatting commands are generated by function keys and menu selections, rather than as embedded markup. Few of these systems today are suited to the batch typesetting required in journal and periodical production, because they bind a graphical input and output interface too tightly to the typesetting machinery; however, that market, because of its publishing volume, will eventually prove attractive.
- Users of most WYSIWYG systems are encouraged by the immediate feedback of the typeset display to make

visual, rather than logical, design decisions. Design professionals often criticize visual design [10, Section 1.4] because it can lead to poor typography. Also, the visual layout may make it difficult to re-use the text, or to reformat it for a different output style. These objections may disappear as newer generations of these systems provide better support for document styles, and separation of the jobs of authoring or document entry, and document design.

- Several desk-top publishing systems are already capable of easily handling multi-column output, multi-column floats, flowing of typeset text around inserts (both rectangular and non-rectangular), and easy integration of graphics with text; these are areas where T_EX is noticeably deficient.

4 T_EX's advantages

In view of the points raised in the preceding section, we must then ask what does T_EX (and I mean also METAFONT, Computer Modern, and related software) offer that competing desk-top publishing systems do not, at least not yet?

- T_EX provides public-domain access to the source code of its related software. Source code of commercial implementations remains proprietary, but the changes from the public domain versions are usually in system-dependent areas that do not affect the overall operation of the software, and for most machines, both public domain and commercial implementations are available.

Public access to the source code is extremely important. It permits both low-cost, or even free, public-domain implementations, and supported commercial implementations, of T_EX to be available on many different platforms. A commercial user of T_EX need not be tied to any single vendor of the software; such ties can become a significant competitive disadvantage when the supplier does not keep up with technological progress. As one such example, I cite the TV Guide experience [1].

Although T_EX is probably one of the most bug-free software packages of its size, it is reassuring to a user to know that if a question ever arises as to why the system typeset text in a particular way, the availability of well-documented source code makes it in principle possible to find the reason. Public access to source code means that bugs are often found and reported by several users, and fixes can come more quickly. By contrast, commercial desk-top publishing systems are almost always unfathomable black boxes whose surprises are indecipherable; it may be difficult to convince a vendor that an anomaly is a 'bug' instead of a 'feature'.

- T_EX source code is written in a relatively portable language, and consequently, it is available *today* for vir-

³WYSIWYG = What You See Is What You Get, sometimes called What You See Is All You've Got.

tually every commercially-available computing system, from personal computers, up to supercomputers.

- The wide availability and use, and the frozen development, of T_EX mean that we can view it as an *archival* document formatting system. Most commercial publishing products have completely ignored this issue; succeeding product generations offer new features and bug fixes, but are often incompatible with earlier ones. It is certainly true that much of what is published today is “throw-away” material, and in such cases, whether the publishing system can reformat the same document years from now is of no concern.

However, in academic circles, this is decidedly not the case. Academicians research and write in the interest of wide dissemination of their ideas, both to current colleagues, and to future generations. Authors and publishers of such material are interested in re-using it for multiple documents. One of the T_EX90 speakers from a major publisher noted that in some fields of study, the same text can be re-used more than a dozen times.

- T_EX’s freedom from architectural and commercial licensing restrictions facilitates collaborative efforts of several authors to work on the same document, even if they have different computer hardware.

- T_EX’s markup is visible, not hidden in magical undocumented binary data embedded in the document. This has several virtues:

- Detection and correction of formatting errors is usually easier when the formatting commands can be seen.
- It is relatively easy to write simple filters that strip the markup from a document to produce raw text which is input to other software tools for spell checking, grammatical analysis, and so on.
- The markup is recorded in the same character set as the raw text, greatly facilitating document exchange between unlike systems, or via electronic mail.

- T_EX’s support for visible markup means that translation may be possible between it and other markup systems, such as SGML-based ones.

- T_EX supports a powerful macro language that permits the creation of separate input interfaces that can be quite different from plain T_EX. *AMS-T_EX* and *L^AT_EX* are the most obvious examples, but the Free Software Foundation’s T_EXinfo and L^AT_EXinfo systems, and the use of T_EX as the typesetting engine for documents written in other markup languages, as is done at at least two major publishing houses, are other examples. Most desk-top publishing systems lack this extensibility.

- T_EX is capable of handling multi-lingual typesetting; few commercial publishing systems today can make this claim.

- T_EX’s mathematical typesetting abilities are still unmatched by most desk-top publishing systems. Its Computer Modern font family, together with the AMS font extensions, provides a repertoire of characters

that is far more comprehensive than almost anything available on other systems. (I was able to announce at the Cork meeting that Adobe Systems has finally released a Lucida font in POSTSCRIPT format with a set of mathematics characters matching Computer Modern. Lucida is the font used in the typesetting of *Scientific American*.) The public-domain nature of T_EX will of course make it possible for commercial systems to incorporate T_EX’s sophisticated algorithms for mathematics; however, this is likely to happen slowly because most of the commercial desktop publishing market has little need for mathematical typesetting.

- T_EX, and other systems based on visible markup (including those that use SGML), have a significant advantage over WYSIWYG systems in that style and content can be clearly separated. In most desk-top publishing systems, style and content are inextricably entwined. This has important ramifications for alternate uses of the input text, for user training, and for the effort needed to change the style without modifying the content.

With T_EX, authors and clerical staff need learn only *one* system that can be used with very minor changes to produce documents in a wide variety of styles.

5 Some observations

T_EX currently has a portability advantage over most other typesetting systems. Many commercial publishing products are tied very closely to the hardware or window system architecture of a specific machine, particularly in the personal computer market. This has meant years of delay in getting them ported to other systems. The rise of the C language, particularly during the 1980s, as an efficient, but nevertheless portable, machine-independent implementation language is slowly beginning to be recognized by vendors. Assembly-language coded systems are now being rewritten in C or C++ to reach a wider market. Recent examples include SAS, Word Perfect, and Lotus 1-2-3. Because of the spread of popular window systems, such as X, Microsoft Windows, and others, and the efforts to standardize them, I expect that by the end of this decade, most commercial software products related to publishing will be available on as wide a range of machines as T_EX currently is.

While it is true that standard T_EX does not provide an immediate visual display of the typeset text, the Berkeley *VORTEX* project, about which too little has been written, and ArborText’s Publisher system are demonstrations that T_EX *can* have such an interface. The rapid advances in computer speeds that have occurred, largely through RISC processor developments, and the volume production economizations possible through sales of millions of personal computers, suggest that we are only a few short years away from instantaneous typeset on-line display.

Few existing systems, including WYSIWYG ones and

T_EX, are suitable for newspaper publishing, which is characterized by its complicated layout of text and graphics in up to six or eight columns, and daily deadlines that cannot be missed without serious economic impact. I expect that the most printing done in the world today is in newspapers. While most of larger newspapers now use computer-based typesetting, I suspect that their systems are rather specialized for that industry.

6 Necessary future developments

The preceding sections have discussed the relative strengths and weaknesses of T_EX versus desk-top publishing systems. I have found in discussions with other TUG members at meetings, and in mail exchanges, that many of us share the view that development of T_EX cannot stand still. Donald Knuth has placed understandable restrictions on the use of the names T_EX, METAFONT, and Computer Modern. Consequently, evolutionary systems arising from T_EX will have to use different names.

I believe strongly that what needs to be done now is for those users of T_EX and METAFONT who have pushed the limits of those systems to begin writing down detailed descriptions of just what those limitations are, and to make well thought-out suggestions about the directions that future work ought to take.

I made a start last year on the relation of T_EX and graphics in [3].

Frank Mittelbach gave a wonderfully incisive exposition on the future of T_EX at the College Station TUG'90 meeting [12], and followed that at the Cork T_EX'90 conference with a fine presentation of work done together with Reinhard Wonneberger on the future of BibT_EX [14].

Michael Vulis has shown with an actual implementation [13] how scalable fonts tightly integrated into a T_EX-like system can offer new and interesting capabilities. To those who would quibble with his incorporation of the name T_EX, I would observe that V_TE_X is a superset of T_EX, and with a special command-line argument, it will disable all extensions and perform exactly like T_EX; nevertheless, it would be advisable to adhere to the Grand Wizard's wishes, and change the name.

John Hobby presented some very promising work at the Stanford TUG'89 meeting on extensions of METAFONT for generation of POSTSCRIPT output [8], and related work by Shimon Yanai and Daniel Berry should soon appear in *TUGboat*.

We need more such articles! Please, if you can contribute new ideas, and I know from personal contacts that many of you can, write them down (or even up) for publication in *TUGboat* or other journals in the field.

Only when we have a solid base of written contributions from the T_EX experts will it be possible for some future researcher to have a reliable starting point for the

design of the evolution of T_EX to the next generation of typesetting system, and that person will have the added challenge of finding new names!

Let us hope that a major design goal of such an effort will be the maintenance of compatibility with existing T_EX and METAFONT input, so that the substantial, and growing, base of existing T_EX and METAFONT material will continue to be processable, with exactly the same results, by the next generation of computer-based typesetting systems. I believe that this would be far preferable to having separate, but mutually incompatible, systems that must try to coexist peacefully.

Incompatibility may eventually become necessary. By the time that T_EX's grandchildren are born, it may be that they will bear little resemblance to their ancestor. We can only hope that use of T_EX will have become so commercially important that translators of T_EX documents to the new generation systems will be developed. An analogy can be found in programming languages: Fortran is a distant ancestor of the Algol family of languages, including Pascal, C, C++, and Ada. An enormous body of important Fortran code exists that cannot possibly be rewritten by hand; public-domain and commercial translators have been developed to convert Fortran code to some of these languages.

While the design of T_EX's children is underway, we need to get all T_EX systems upgraded to the final versions that Donald Knuth has provided, and we need to agree upon a standard 8-bit T_EX font encoding that will permit the exchange of documents that make use of the new features of T_EX 3.0. As I noted in my President's message in this issue, this second problem should soon be solved.

References

- [1] Elizabeth Barnhart. T_EX in the commercial environment—multi-column output. *TUGboat*, 8(2):185, July 1987.
- [2] Zeev Becker and Daniel Berry. `tri`off, an adaptation of the device-independent `troff` for formatting tri-directional text. *Electronic Publishing—Origination, Dissemination, and Design*, 2(3):119–142, October 1989.
- [3] Nelson H. F. Beebe. T_EX and Graphics: The State of the Problem. *Cahiers GUTenberg*, 1(2):13–53, 1989. Presented to: Congrès GUTenberg, Paris, France, 16–17 May 1989.
- [4] Michael Ferguson. Multilingual T_EX update. *TUGboat*, 7(1):16, March 1986.
- [5] Michael Ferguson. A (hopefully) final extension of multilingual T_EX. *TUGboat*, 8(2):102, July 1987.
- [6] Michael Ferguson. Coordination of non-English use of T_EX. *TUGboat*, 11(1):8–9, April 1990.

- [7] Michael J. Ferguson. A multilingual T_EX. *TUGboat*, 6(2):57, July 1985.
- [8] John D. Hobby. A METAFONT-like System with PostScript Output. *TUGboat*, 10(4):505–512, December 1989.
- [9] Donald Knuth and Pierre MacKay. Mixing right-to-left texts with left-to-right texts. *TUGboat*, 8(1):14, April 1987.
- [10] Leslie Lamport. *L^AT_EX—A Document Preparation System—User’s Guide and Reference Manual*. Addison-Wesley, 1985.
- [11] Franklin Mark Liang. *Word Hy-phen-ation by Com-pu-ter*. PhD thesis, Stanford University, August 1983.
- [12] Frank Mittelbach. E-T_EX: Guidelines for future T_EX extensions. *TUGboat*, 11(3):337–345, September 1990.
- [13] Michael Vulis. V_TE_X extensions to the T_EX language. *TUGboat*, 11(3):429–434, September 1990.
- [14] Reinhard Wonneberger and Frank Mittelbach. B_IB_TE_X reconsidered. *TUGboat*, 12(1):xxx–xxx, March 1991.

