BIJLAGE U

EuroTEX 89

Karlsruhe, 11-13 sept 1989

J.L. Braams en C.G. van der Laan

LeX: Polish TEX

Abstract

Algemeen
- Polen stal de harten

Organisatorisch
- Veranderingen LATEX, gecoordineerd door Frank M. en Jan R.
- Teaching TEX in discussie.
- Cork bijeenkomst voorbereid
- Zowel PD als commerciële TEX aktiviteiten nodig
- Een Nederlandse vertaling van Schwarz's Einführung in TEX is op komst
- Leerzaam handout voor potentiële DANTE leden, inclusief questionnaire
- Quixote Oriental Fonts project (Hosek)

Producten
- Schroder's changebars macro
- Thull's MS-DOS PD TEX
- TEX-omgevingen: PCTEXE (MS-DOS), TEXE (AOS/VS), CEO-TEXE (Data General)

1 Reizen

Vanwege de ongelukkige aankomsttijden van de treinen hebben wij besloten met de auto te reizen. Op het laatste moment vonden wij elkaar via TeX-NL. In Karlsruhe bleken er nog twee Nederlanders te participeren: Andre de Meijer en Ton Biegstraaten.

2 TUG zaken

Er was afgesproken dat de kersverse nationale TUGbobo's samen met Peter F, Bart C en ReG, de Cork bijeenkomst zouden doorspreken, alleen was er geen plaats genoemd. Geen nood, even rondfietsend ontdekte ik Cathy en Malcolm op een terrasje en na een hartelijke begroeting voorafgegaan door verbazing — Kees op de fiets?! — werd het duidelijk dat er op het terrasje vergaderd zou worden. Heel genoeglijk en goedkoop! Voor de mannen, onder hilariteit van Cathy, was het wat moeilijk de aandacht bij de les te houden vanwege de flanerende dames.

3 De conferentie

Allereerst enige 'omtoe'-sessies: Status TUG, status Teaching TEX, DANTE's bestaan, NTG's ontstaansgeschiedenis, de situatie van TEX in Europa en de organisatie er omheen. Dit is allemaal min of meer al ergens anders uit de doeken gedaan: TUGboat (Teaching TEX), Stanford verslag (Status TEX in Europa), en CGL's voordracht voor de NTG dagen en alhier.

Fileservers

Ook de sessies over het gebruik van de fileservers gaf geen nieuws. De discussie er omheen wel: Hoe er mee om te gaan? Minimale documentatie? Datering? Taxonomy en inhoudsopgave? Specialisatie? Intelligente server? e.d. Ook dit is elders al aangestipt, zie GUTenberg verslag.

Polish TEX

Heel charmant en kundig was de Poolse bijdrage over Polish TEX. Alom werden de harten gestolen: TUG en iedereen is bezig om Polen de helpende hand te bieden. De NTG overweegt Ryćko op de TEX–SGML bijeenkomst uit te nodigen, waarbij wij logies (privé), 'het ontmoeten der mensen' en het verder proeven aan de westerse samenleving kunnen aanbieden.

Relatie EP

De relatie met SGML en in het algemeen het Electronisch Publiceren kwam aan de orde via de verhalen van Malcolm C, David R, Martin B, en Anne B-K. Malcolm was goed op dreef: inhoudelijk leuk, niet te technisch en vooral uitdagend. Zijn belangrijkste punt was dat vorm en inhoud niet strikt gescheiden kunnen worden: de vorm kan de boodschap significant versterken; slechts de auteur weet waar en hoe. Conclusies:

- SGML needs TEX
- TEX can learn something from SGML
- Documents need formatting
- The output medium must be known
- Formatting requires aesthetic decisions
- Contents and structure are intertwined

David Rogers, Arbortext, gaf zijn visie op het EP gebeuren. Hij denkt aan een werkomgeving met sjablonen en ikonen. Achter deze werkomgeving zitten respectievelijk een SGML, LATEX en Postscript laag, waarbij het mogelijk is de tussenresultaten ook te verkrijgen, dus de SGML , LATEX of Postscript file. Ook was het intrigerend dat zijn benadering hoge ogen gooit in de CALS tender, waarbij giganten, zoals IBM, gepasseerd worden!

Toekomst LATEX

Parallel aan de lezingen werden er ook discussie bijeenkomsten georganiseerd. De sessie van Frank M en Reiner S over de toekomst van LATEX was heel relevant. Hij besprak met de aanwezigen een wensenlijstje, dat na toetsing op relevantie en realiseerbaarheid, met Knuth's TEX veranderingen meegenomen zou moeten worden. De volgende punten resulteerden

- restart paragraph building (with modified parameters)
- system command usage
- dynamic tfm file
- more (>32) font families
- appending to a file (`\openappend`)

Deze punten geadstrueerd met een overtuigend voorbeeld voor elk punt zouden zo snel mogelijk naar DEK opgestuurd worden onder c.c. naar de aanwezigen.
Opmerking: Tot op heden nog niets ontvangen. Drukte alom?

TEX in het middelbare onderwijs

Charmant en ook indrukwekkend was de ervaring van een middelbare scholier, Neuwirth, met TEX op een Amiga. Juist Duitsland profileert zich t.a.v. het lanceren van TEX in het middelbare onderwijs. Just wait and see.

Conclusie

Een nuttige bijeenkomst waarbij de contacten zijn verstevigd en de samenwerking verder is gegroeid. DANTE is een soliede vereniging met

- $\pm$200 leden,
- een fileserver,
- listserver,
- een grote achterban met leden zoals Frank M,
- een eigen tijdschrift (toekomst),
- auteurs van goede TEX boeken (Appelt, von Bechtolsheim, Schwarz, Wonneberger).

Bovendien hebben zij een keur aan TEXhackers die ook nog les kunnen geven. Een machtige, goede buur! Aan de andere kant is er een generatiekloof te bespeuren tussen de jongedienden van DANTE en de wizards van het eerste uur: Waarom bijvoorbeeld is de DANTE bijeenkomst niet simultaan met deze EuroTEX89 gehouden?

Maar, . . . , wat een perspectief ligt er nu 'het gordijn is neergehaald'!

References

[1] Antweiler,W., A. Strotman, V. Winkelman (1989): Typesetting REDUCE output with TEX. Rechenzentrum Universität Köln.

[2] Brüggemann–Klein, A. (1989): Einführung in die Dokumentenverarbeitung. Teubner.

[3] DANTE Deutschsprachige Anwendervereinigung TEX e.V. Antragsformular zur Mitgliedschaft.

Bijlage A: Teaching TEX: Some Critics and LATEX proposal.

BIJLAGE V

Teaching TEX: Critics & LATEX proposal [1]

C.G. van der Laan

November, 1989

Abstract

Some critics — mainly LATEX based — on the approach are given: TEXnically driven instead of result driven. Also is notified the omission of a general pedagogical method to be used. Furthermore, it is urged to have unified courseware. A set of LATEX courses is proposed. No test set is included. The underlying idea of having TUG certified courses world-wide is strongly endorsed.

1 Why?

It has been on my desk for so long and I could not come to make my ideas more explicit than the oral comments given in Karlsruhe. This can be explained partly from the fact that I disagree with the TEXnically driven approach for the LATEX courses. Moreover, I consider it worthwhile to provide TEX courses also from the demand driven viewpoint. Therefore I was hesitant and in doubt about the effect in comparison to the invested energy. But we are TEXies are not we, and therefore we do what we think is useful and don't worry about the human costs but are led by the beneficial effect of exchanging ideas, building upon each other, and sharing the outcome!

2 Classification

2.1 Endorsement

Agreed the tests are clear and simple to apply within the TEX field, i.e., neglecting other experience. Happily, the answers are also published, seen [3].

2.2 But ...

I have some problems with the classification from the point of view as coursetaker. What do beginner, intermediate and advanced really mean? I don't overlook that a tool is provided to give the answer for every particular case. The point I like to stress is that beginner etc. do not account for general computer science experience. I mean, taking myself as a beginner — in any CS related field — is different from a young enthousiastic personnel member, who likes to know what (La)TEX is about. This theme is elaborated below.

Let me make this clear by personalizing this. I don't know whether I'm a beginning TEXie, intermediate or, well ... certainly not advanced. From the other way round I know: I don't consider myself as advanced, therefore I have to choose between beginner and intermediate. Honestly speaking I'm a beginner, because I partly read the TEXbook, understood part of what I read, and remembered hardly nothing, and well ..., the test set I could make with 75% correct, I hope. But in programming (at least the principles of), in mathematics, in electronic publishing, making publications, handling SGML, to name but a few, I'm not a beginner, although for the precise coding I have to look at manuals, because I'm confused by the variations and have given up to remember the details of ALGOL60, PASCAL,

[1] This note is a contribution to the discussion guarded by Bart Childs, see TUGboat and/or the Karlsruhe EuroTEX89 proceedings.

ALGOL68, ADA, FORTRAN, BASIC, C and even dawning WEB, apart from OS details of VMS, NOS/VE, UNIX, MS-DOS, ... And then there are other packages and the huge (numerical) libraries ...

Is it not true that 'beginners' are invariantly burdened with things they already know from general experience? Therefore, I estimate this classification will entail a nonhomogeneous audience, with its pedagogical difficulties. These difficulties can be lessened by an individual approach in the 'laboratory' part, but then a superset of exercises have to be provided where the student can choose from.

2.3 Classification pragmatics

Let us adopt the given scheme for TEX, but let us adopt for LATEX a demand driven approach. By the way the metafont announcement did sound demand driven: design logos and font design! This approach for LATEX is endorsed by the experience gained in other fields as well: driving cars, using washing machines, radio, TV etc. People make use of it without understanding it except possibly for the principles. They are not burdened with the TEXnicalities why it works. If 'AS IS' does not work any longer a TEXnician is called. See, a.o., also Kubek's TEX for the wordprocessor operator, presented at Stanford. Moreover, learning to drive a car is not done by a mecanicien but an instructor.

3 Pedagogical principles

I missed the mentioning of pedagogical principles used. Not only *WHAT* is taught is important but also *HOW*! Especially, if we strive after worldwide TUG courses, given by various teachers, it is important to *teach the teacher*, not only with respect to TEX but the more so with how to teach. Train them in the pedagogical principles to be used. I mentioned in Karlsruhe that in nonregular education, I'm also involved with, this is done for a hundred or so free-lance teachers who are trained in this spirit. They are supported with recognizable courseware, such that like a book, anybody can smell from a rather remote distance that this course is from company so and so, without paying attention to the contents, just from form principles. This holds for the same courses given at different places by different teachers as well as for related courses. You undoubtedly are aware of the Gagné principles, which try to break with just listening or hands-on or both, but also pay attention to rehearsing, removing blockades, clearifying techniques and the like.

4 Courseware

The homogenity of the same course given at different places by different teachers will strongly benefit from unifying supporting courseware. The test set is the first step in this direction. The next step concerns the courseware. For courseware we need syllabi as hand-outs, related transparencies, sets of exercises and answers. For exercises I would recommend a superset where the teacher and especially the student can choose from, given their background and interest. Also an 'open' exercise, the problem the 'students' are working on at home, is allowed. These aspects, although sound for themselves, also alleviate the process in a nonhomogeneous class.

The development of the courseware could be organised as well. Either centrally organised and subsidised by TUG or via examples/pilots based on the specs which are in discussion now, and controlled by let us say TUG once again. The development of courseware is an order of magnitude more laborous then translating developed (English) courseware into other languages.

5 LATEX proposal

Invariantly I ask myself the question for whom I'm teaching and what is the (teaching) goal. The first is important because it makes clear *where to start*, in other words what is already known. This sounds trivial but of what I have seen, this is generally not adhered to. From highschool to university it is invariantly neglected in the Netherlands. The teachers don't start in the 'language', or from the situation, the students are familiar with. With our LATEX course I aimed at report writing for scientists, i.e., the coursetakers should be able after the course to make a report which contains ordinary text, formulas, illustrations and some tabular matter all embedded in the general report structure, with title matter, foreword, table of contents, chapter, (sub ...)section, paragraph structure, appendices and end matter such as index, glossary, references and the like, embellished with exceptions like end/footnotes, margin remarks and cross-referencing. Assumed experience: can handle an editor, access a LATEX computer, can make some publication. Moreover it is assumed that the contents of the publication is understood.

After having stated the aim I try to abstract from the tool (LATEX in this case). The exercises were brought together independent from the limitations of LATEX and prompted by practical needs, well ..., that is what I urged my co-authors to do. The consequence of this is that emphasize is put on *WHAT* to accomplish – demand driven — and less on *HOW TO* — technique driven. This makes it also useful in other contexts. It is hopefully abundant to say that in a course (Report)Publishing with LATEX the 'HOW TO' must be explained in the chapter preceding the exercises. The answers are of course within the context of the tool.[2] Note the charming side effect of this approach: exercises which cannot be handled easily, or not at all, are allowed. In our interface examples of usage of FORTRAN libraries in HLP's, some examples are worked out with the comment: Sorry, using FORTRAN routines with a procedure as parameter are not possible from PASCAL. Thís elaboration is time (place or manufactory) dependent, while the set of examples, judiciously chosen, suffer much less from 'wearance'.

Personnel vs. scientists

This paragraph is not aimed at polarisation between personnel and scientists. For personnel the same exercises could be given, i.e., the same results are wanted.
More attention must be paid to the way of working, i.e.,

- 1^e personnel are used to the WYSIWYG method of working (so a sun-like configuration is wanted, or provide a user interface they are familiar with (e.g., Anita H's WP interface)),
- 2^e personnel do not necessarily understand the contents, and
- 3^e personnel are doing this kind of work most of the day.

At the moment, what we call, β-scientists, are familiar with programming and therefore with the command approach, but most of all they know where the publication is about and understand every detail. This different attitude must be accounted for.

5.1 Demand driven

With the above approach it is easier to adapt EP courses to say SGML, (Report) Publishing with SGML, or how to achieve Publishing with TEX, or even WP. This approach also allows to use different techniques with the same printed result, e.g., a form can be obtained via the tabular and picture environment. At our University I invariantly shake my head about LATEX and WordPerfect classes differing so much in approach and courseware, and that the teachers hardly communicate. Even the next WP teacher complete disagrees with his predecessor and develops once again courseware, generally ingeniously motivated. This demand vs. technique driven approach also emerged in the/my teaching of programming languages, where the principles of programming are kept in paradigms and a certain discipline, while the coding is determined by — here and now — suitable compilers.

[2] Within the context of numerical mathematics this principle led to practice on any computer familiar to the students; no restriction to VMS, UNIX, NOS or whatever.

5.2 (LATEX) courses

From the practical point of view I will restrict myself for the moment to a set of LATEX courses, but such that we could substitute SGML, TEX, Wordperfect, ..., for LATEX and still have a practical set.

- LATEX an introduction, or beginning LATEX.
  One day introduction. An overview what it is all about, where to get, hardware requirements, how to get macros, information sources (TUGboat, TEXHaX, ...), how it is supported (use of listservers: TEXHaX, and national listservers), estimated lifetime, ease of learning, survey of courses which can be taken, documentation, relation to alternatives.
  Scheme:
  2 hours introduction (general aspects) and letter as example,
  coffee break
  1 hour hands-on: personalizing letter template
  lunch
  1 hour article introduction
  2 hours letter hands-on (personalizing template).
  Result: as appetizer a letter and article are produced starting from templates, so the class will have something to take home, and not only have heard about the quality but also experienced it.
  Prerequisite: some (Personal) computer experience.
- LATEX Wordperfect user interface.
  One day. How to use Anita H's macros. (This course is site dependent. At our site Wordperfect is the standard wordprocessor for the moment.)
  The knowledge of the introductory course is assumed.
  Prerequisite: experience with document structures, using a wordprocessor, using computers, using fileserver(s).
- Publishing with LATEX.
  Five days. Only one style (report) is dealt with 'AS IS'.
  Scheme:
  Every morning theory and every afternoon laboratory work.
  1^{st} day: introduction and ordinary text (structure, local exceptions: end/footnotes, margin remarks),
  2^{nd} day: mathematics,
  3^{rd} day: tabular matter,
  4^{th} day: graphics, and fonts,
  5^{th} day: integration (toc, references, index, headers/footers, title matter, ...).
  Result: a report and experience how to make details of it. The knowledge of the introductory course is assumed.
  Prerequisite: experience with writing a report, handling an editor, using computers, using fileserver(s).
  Note. The $2^{nd}, 3^{rd}, 4^{th}$ day can be interchanged or deleted dependent on the circumstances, a modular approach.
- More Publishing with LATEX.
  Three days. More a workshop. Advanced LATEX tools which have been developped by the community.
  (Drop macro, bridge, chess, changebars, endnotes, tables over several pages, more column formats, bezier style, trspar style, ...)
- Modification of sty-files.
  Three days. Given a GENERIC sty-file the various modifications are explained and instantiated. The exercises build up to instantiation of a specific `journal.sty`.
  The experience gained in the Publishing with LATEX and intermediate TEX courses are assumed.
  Prerequisite: experience with writing a report, handling an editor, using computers, using fileserver(s).

Note. No scheme is provided as of yet. Even the precise number of days is not fixed.

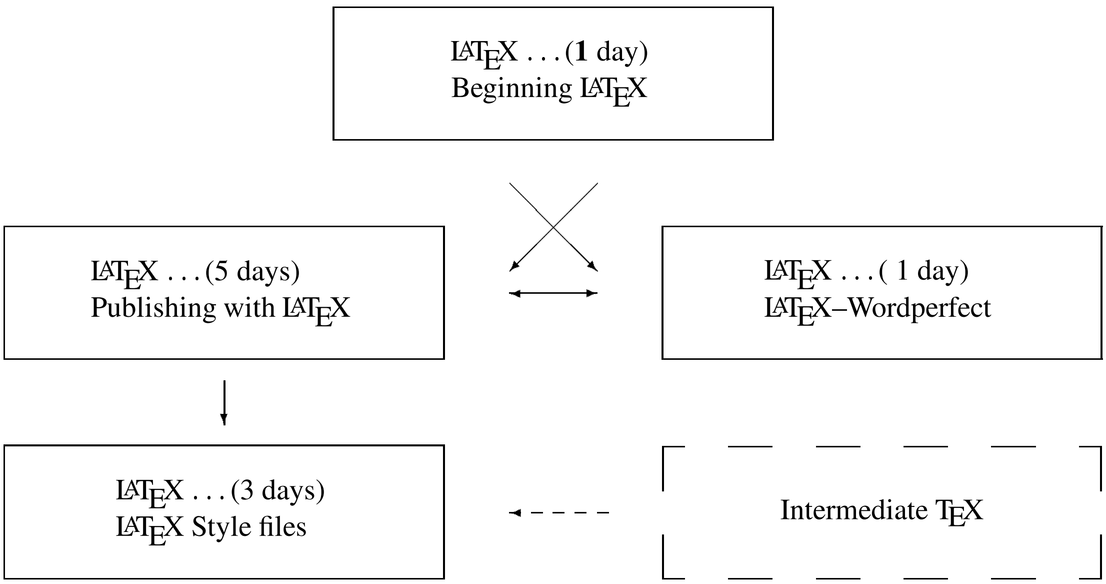
Remark. Courseware to start from is available for:
Beginning LaTeX: Letter and article templates, RC-RUG report 27.
Publishing with LaTeX: Publiceren met LaTeX. CWI-syllabus 19.
LaTeX style files: Journal.sty guidelines, RC-RUG report 26; or, Documentstijlontwikkeling. TeXnique. Utrecht.

5.3 (self)Test set

No test set is included. If the approach is adopted, I would also like to contribute to the test set.

5.4 Dependence

For advanced LaTeX style course the TeX intermediate course must also be a requirement. I missed this interrelationship in the course scheme, [2]. In the scheme below the More Publishing with LaTeX course is not displayed. Perhaps it could best be sketched in the third direction.



Conclusion

It is hoped that this paper will contribute to an international agreed upon set of TeX and LaTeX courses.

References

[1] Bruin, R. de, C.G. van der Laan, J.R. Luyten, H.F. Vogt(1988): Publiceren met LaTeX. CWI-syllabus 19. Centrum voor wiskunde en informatica. Amsterdam.

[2] Childs, B. (1989): Teaching TeX. TUGboat, 10#2, 156–163.

[3] Childs, B. (1989): Answers to TeX tests. TUGboat, 10#3, 319–323.

[4] Doob, M. (1989): A gentle introduction to TeX. A manual for selfstudy. (Also as RC-RUG rapport 25).

[5] Laan, C.G. van der, J.R. Luyten (1989): Letter and article templates. RC-RUG report 27.

[6] Poppelier, N. (1989): Documentstijlontwikkeling. TEXnique. Utrecht.

[7] Steemers, L., C.G. van der Laan (1989): Journal Style Guidelines. RC-RUG report 26.

BIJLAGE W

Wat is TEX?

Victor Eijkhout en Nico Poppelier

versie 0.8 – 20 december 1989

Abstract

In de huidige tijd van WYSIWYG ‘desktop publishing’ systemen lijkt het vreemd een systeem te propageren met honderden, vaak ingewikkelde, instructies. Wat het tekstopmaaksysteem TEX desondanks aantrekkelijk maakt, is het feit dat het programmeerbaar is: de basisinstructies kunnen worden samengevoegd tot zeer gecompliceerde instructies die een eenmaal ontworpen opmaak geheel automatisch feilloos reproduceren. Hierdoor hoeft de gebruiker uitsluitend nog de structuur van het document aan te geven en niet de vormgeving, het uiterlijk, ervan. Met deze gedachte sluit TEX nauw aan bij recente ontwikkelingen in de uitgeverswereld omtrent SGML.

1 Inleiding

TEX[1] is een tekstopmaaksysteem met een wat merkwaardige onstaansgeschiedenis. Het is ontwikkeld door Donald Knuth, een hoogleraar computerwetenschappen aan de universiteit van Stanford in Californië. Terwijl hij bezig was met het schrijven van zijn boekenreeks ‘The Art of Computer Programming’ kreeg hij ideeën voor een ‘programmeertaal voor tekstopmaak’. Tussen 1978 en 1984 groeide TEX zo geleidelijk tot zijn huidige vorm.

Het is, toegegeven, een merkwaardig idee dat een softwareprodukt zich al meer dan zes jaar niet meer ontwikkeld heeft. Toch betekent dat niet dat TEX verouderd is; veeleer was het zijn tijd ver vooruit toen het ontstond. In die jaren is overigens slechts een handjevol fouten in het programma gevonden.

Er zijn meer ongewone aspecten aan TEX. Zo is er voor practisch elk type computer wel een versie van TEX, en vaak zelfs een gratis, een zogenaamde ‘public domain’, versie. De uitvoer van TEX is al net zo universeel: voor bijna elke laserprinter en fotozetter is er wel een programma dat door TEX opgemaakte teksten kan afdrukken, ongeacht op welke computer het opmaken is gebeurd.

Bij veel mensen die wel eens van TEX gehoord hebben, heeft het idee post gevat dat het een moeilijk systeem is, alleen te bedienen door computerprogrammeurs. Dit misverstand is begrijpelijk gezien de ontstaansgeschiedenis van TEX. Het is echter niet juist.

De waarheid heeft twee kanten. Enerzijds heeft TEX ongewoon veel opdrachten, enkele honderden, wat meer is dan gewone zetsystemen hebben, en veel meer dan gewone programmeertalen. Ook is de structuur van TEX gewoonweg lastig in vergelijking met normale zetsystemen en programmeertalen. Anderzijds kan iemand die bedreven is in TEX de ‘primitieve’ opdrachten van TEX samenvoegen tot nieuwe opdrachten op een hoger niveau, ‘macro’s’ genaamd, die zeer gecompliceerde dingen doen. Hiermee kan het bedienen van TEX dus willekeurig eenvoudig worden gemaakt.

2 Wat kan TEX zoal?

TEX heeft in zijn simpelste vorm een aantal elementaire zetinstructies ingebouwd als ‘zet de volgende tekst cursief’, ‘geef tien punt witruimte’ en ‘gebruik vrije regelval’. Omdat TEX geprogrammeerd kan worden, kunnen veel ingewikkelder instructies dan dit gerealiseerd worden.

[1] Uitgesproken ‘tech’; dit is de hoofdlettervorm van $\tau\epsilon\chi$; de officiële schrijfwijze wanneer de gezakte ‘E’ niet mogelijk is, is `TeX`.

Een voorbeeld: iemand zou een macro 'paragraaf' kunnen maken zodat een gebruiker intikt

```
\paragraaf Voordelen van TeX
```

en daarmee de volgende serie handelingen veroorzaakt:

- er komt een zekere hoeveelheid witruimte achter de voorafgaande tekst;
- het systeem houdt een 'paragraafteller' bij en drukt de waarde daarvan af, gevolgd door de tekst van de paragraafkop, in een zeker lettertype en -grootte; daarna wordt de paragraafteller met één vermeerderd;
- de titel wordt, met het bijbehorende nummer en met het paginanummer, automatisch aan een door het programma gegenereerde inhoudsopgave toegevoegd;
- als er kopregels gewenst zijn, wordt de paragraaftitel daar automatisch in ingevuld;
- achter de titel komt wat witruimte, met die verstande dat TEX nooit de pagina zal breken tussen de titel en de daaropvolgende tekst – al naar gelang wat van te voren besloten is kan de witruimte boven de titel wat samengedrukt worden, of kan de titel naar de volgende pagina verhuizen, waarbij de huidige pagina mogelijk opgerekt wordt om de onderste regel te laten aanlijnen met andere pagina's;
- de eerste alinea na de titel springt niet in, maar alle daarop volgende wel.

Het valt zelfs zo te regelen dat de gebruiker intikt

```
\paragraaf Voordelen van TeX\referentie{voordelen}
```

en dat hij later kan schrijven

```
... zoals betoogd in paragraaf \refereer{voordelen}
```

waarbij TEX dan automatisch het juiste nummer van de paragraaf invult.

Verschillende mensen over de wereld hebben al uitgebreide pakketten van dit soort 'hoog-niveau opdrachten' voor TEX gemaakt. Het bekendste pakket – in TEX termen: een *format* – is LATEX, dat uit 1985 stamt en is ontworpen door Leslie Lamport, een medewerker van *Digital Equipment Corporation*. LATEX is gebaseerd op het idee van beschrijving van de logische structuur van een document: de opdrachten die in de tekst staan geven informatie over de *functie* van onderdelen van de tekst. De vertaling van de logische structuur naar een visuele structuur is vastgelegd in zogenaamde 'documentstijlen'. De gebruiker moet aan het begin van een document eerst aangeven 'dit is een brief' of 'dit is een artikel', en afhankelijk hiervan staan hem opdrachten ter beschikking. Zo bevat de artikel-stijl wel de opdracht 'titel', maar de brief-stijl niet.

Over het algemeen geldt dat wat TEX nog niet kan, hem geleerd kan worden. Hier is echter wel wat vaardigheid voor nodig. In het tijdschrift van de TEX-gebruikersgroep verschijnen regelmatig oplossingen op allerlei zetproblemen. De eerste auteur van dit artikel heeft bijvoorbeeld 'variozetsel' en 'hangend inspringen' op zijn naam staan.

Los van deze toeters en bellen is het resultaat van TEX op het gebied van gewone, 'platte' tekst vaak al beter dan dat van andere systemen. De reden hiervoor is het feit dat TEX niet per regel werkt maar per alinea: pas als een complete alinea af is wordt bepaald waar de regels gebroken gaan worden. Aangezien TEX ook een bijzonder goed woordafbrekingssysteem heeft, leidt dit proces tot zeer egale alinea's. Overigens vallen de beslissingen van TEX hoe de alinea gebroken moet worden op velerlei wijze te beïnvloeden.

Op het gebied van woordafbreking heeft TEX nog meer in huis: er zijn voorzieningen om het Duitse woord 'backen' te laten afbreken als 'bak-ken', en om de trema in 'onbeïnvloedbaar' te laten verdwijnen in 'onbe-invloedbaar'.

3 De commerciële status van TEX

Het draagt bij aan de vreemde status van TEX dat het in zeker opzicht een 'gratis' pakket is. Voor enkele, op universiteiten veel voorkomende, typen computers is tegen kostprijs een magneetband met TEX te krijgen, en ook voor microcomputers als MS-DOS-computers, de Apple MacIntosh en de Atari ST zijn gratis versies te krijgen. Aan deze gratis versies zijn meestal twee problemen verbonden: één waar ze te krijgen, en twee, hoe ze te installeren. Er zijn echter ook commerciële versies van TEX, die vaak meer gebruiksgemak bieden of sneller zijn.

Een aanzienlijk deel van het zetwerk dat momenteel met TEX gedaan wordt is duidelijk als zodanig te herkennen. Dit is grotendeels te wijten aan het 'gratis' karakter van TEX. Het enige lettertype dat namelijk normaliter meegeleverd wordt, is een kopie van de Bodoni-achtige 'Modern 8A' van het bedrijf Monotype. Er is echter een bedrijf dat een groot aantal ITC-lettertypes voor TEX levert, zoals Times Roman, Baskerville, Bembo en Gill Sans. Ook zijn er bedrijven die lettertypes voor niet-Westeuropese talen leveren, die kunnen worden gebruikt met TEX.

Het zetwerk van het LATEX-pakket is meestal ook eenvoudig te herkennen als de gebruiker de standaard meegeleverde documentstijlen heeft gebruikt, die nogal 'Amerikaans' zijn. LATEX is echter zeer zeker niet synoniem met deze opmaak, zoals de beide auteurs van dit artikel bewezen hebben. De eerste auteur heeft, met advies van een typograaf, documentstijlen ontwikkeld die dezelfde opdrachten bevatten als de LATEX documentstijlen ('artikel', 'rapport', 'boek' en 'brief'), maar met een meer Nederlandse opmaak; de tweede auteur heeft, in opdracht van een grote uitgever, documentstijlen ontwikkeld die exact de opmaak van reeds bestaande tijdschriften reproduceren.

4 Wie gebruikt TEX?

Gezien de ontstaansgeschiedenis van TEX is het weinig verbazend dat het eerste antwoord op de vraag 'Wie gebruikt TEX?' luidt: wiskundigen, natuurkundigen, sterrenkundigen en informatici. Het wordt hen dan ook erg aantrekkelijk gemaakt TEX te gebruiken, aangezien steeds meer tijdschriften artikelen in TEX-vorm accepteren – en dat vaak belonen met versnelde publicatie.

Ook buiten de wetenschappelijke wereld begint TEX de laatste tijd geaccepteerd te raken. Zo werkt de Amerikaanse 'TV-Guide' – wekelijks meer dan honderd verschillende edities, met een gezamelijk omvang van ruim 10 000 pagina's – al geruime tijd met TEX. Dichter bij huis gebruikt het Energiebedrijf voor Groningen en Drente TEX voor het drukken van handleidingen.

In beide gevallen garandeert de macro-faciliteit van TEX dat, zonder dat de mensen die de teksten voor bijvoorbeeld handleidingen intoetsen een grafische opleiding gevolgd hebben, de uitvoer geheel uniform in opmaak is. Voor documenten zoals handleidingen, waarvan geregeld nieuwe versies verschijnen, is de bovengenoemde mogelijkheid om door middel van symbolische sleutels aan secties en andere genummerde onderdelen van de tekst te kunnen refereren natuurlijk bijzonder waardevol: het betekent dat verwijzingingen altijd correct zijn, zonder dat ze ooit gecontroleerd hoeven worden.

5 TEX in het gebruik

TEX lijkt beduidend meer op klassieke zetsystemen dan op de moderne 'desktop publishing' programma's. Eerst moet een tekst ingetoetst worden, besprenkeld met opdrachten, en pas nadat de computer deze invoer bewerkt heeft kan het resultaat op het scherm of op papier bewonderd worden.

Omdat TEX niet echt visueel werkt, is het voor toepassingen als advertentie-opmaak minder geschikt. De grote kracht van TEX ligt echter in de mogelijkheid een bepaalde, vooraf vastgelegde opmaak geheel te kunnen automatiseren. Enkele voorbeelden: een tijdschrift, een boekenreeks, handleidingen of brochures van een bedrijf, de concertprogramma's van een impressariaat of een regelmatig opnieuw verschijnende prijslijst.

Terwijl tot nu toe een grafisch opgeleid persoon zich strikt aan bepaalde richtlijnen moest houden om een reeks publicaties een uniforme opmaak te geven, kan een competent TEX-programmeur deze opmaak in een aantal eenvoudig te gebruiken macro's verstoppen.

Ten eerste vermindert dit de kans op fouten aanzienlijk en ten tweede betekent het dat er niet meer continu een graficus nodig is. Zelfs voor gebruik binnen een grafisch bedrijf zijn er voordelen aan het gebruik van TEX: voor een opmaak die meerdere malen gebruikt moet worden werkt het produktieverhogend, terwijl bij eenmalige documenten de auteur makkelijker correcties aan kan brengen.

De vraag waar iemand de kennis vandaan haalt om TEX-macros te ontwikkelen, kan beantwoord worden al naar gelang de behoefte. Zo zijn er grote bedrijven die hun eigen 'TEXpert' in huis hebben. Aan de andere kant van het spectrum, waar bij voorbeeld wetenschappers hun eigen artikelen in TEX zetten, bestudeert men de boeken over TEX en LATEX, en leert men al doende.

Tussen deze uitersten zitten nog enkele mogelijkheden: er zijn cursussen in TEX en LATEX, al dan niet gegeven onder auspiciën van de internationale TEX Users Group, en er zijn free-lance TEX-programmeurs die hun diensten aanbieden voor het ontwikkelen van LATEX documentstijlen of gespecialiseerde macros.

6 TEX als 'engine'

Voor veel programma's, zoals databases, geldt dat het uitvoer verzorgende gedeelte redelijk schamel is. Per slot van rekening is een database-programma in de eerste plaats ontworpen om gegevens op te slaan, niet om deze af te drukken. De programmeerbaarheid van TEX heeft tot gevolg dat het systeem bij uitstek geschikt is om als 'print engine' voor bij voorbeeld een database te fungeren. Als de rapport-generator van een database-programma, of een ander uitvoer genererend programma, in staat is TEX-instructies tussen de gegevens te plaatsen, kan de kracht van TEX benut worden om de uitvoer op elke gewenste manier af te drukken.

Hoewel wij mensen kennen die zelf hun database-programma zo ver gekregen hebben dat het TEX macros in de uitvoer zet, wordt deze toepassing van TEX nog vrij weinig door producenten van programmatuur zelf toegepast. Als voorbeeld kennen wij wiskundige pakketten waarmee gecompliceerde formules berekend kunnen worden; het pakket kan dan de TEX-instructies voor een dergelijke formule opleveren. Een ander voorbeeld is het Amerikaanse computerbedrijf Digital Equipment Corporation, dat programmatuur gebruikt voor het schrijven en onderhouden van handleidingen, waar TEX als afdrukmachine achter zit. De persoon die de handleidingen intoetst krijgt echter nooit iets van TEX te zien.

Dergelijke toepassingen zullen in de toekomst waarschijnlijk meer gaan voorkomen, nu de uitgeverswereld de ISO-standaard SGML voor documentproduktie lijkt te gaan aanvaarden.

7 TEX en SGML

Het idee van gestructureerde documentopmaak, waarop onder andere, maar niet uitsluitend, LATEX is gebaseerd, is in de meest verregaande vorm uitgewerkt in de door de standaardisatie-organisatie ISO vastgelegde taal SGML. SGML wordt gebruikt of zal gebruikt gaan worden door de Amerikaanse ministeries van Defensie en Economie, de Amerikaanse uitgevers, de Europese Gemeenschap, het onderzoeksinstituut CERN in Genève en een aantal Europese uitgevers, waaronder *Elsevier*. Het laat zich aanzien dat SGML in de komende decennia een steeds grotere rol zal gaan spelen en dat langzaam maar zeker op SGML gebaseerde systemen de huidige WYSIWYG-systemen zullen gaan overvleugelen of dat op SGML gebaseerde systemen met een WYSIWYG-karakter op de markt zullen worden gebracht.

Ook binnen de gemeenschap van TEX- en LATEX-gebruikers is er over SGML nagedacht. Het basisidee van SGML is gegeneraliseerde, dat wil zeggen structuurbeschrijvende, in plaats van uitvoergerichte, zogenaamde descriptieve, 'markup'. In verschillende opzichten is dit idee al gerealiseerd in LATEX, of realiseerbaar in TEX. Doordat TEX volkomen programmeerbaar is, kunnen macro's geschreven worden die bijna als directe vertaling van SGML-instructies kunnen fungeren. TEX lijkt hierdoor het perfecte 'back end' voor een op SGML gebaseerd tekstopmaaksysteem.

8 Besluit

Onze ervaring is dat TEX een flexibel en uitstekend bruikbaar systeem is. TEX heeft de volgende voordelen:

- TEX levert zetwerk van superieure kwaliteit.
- Het kan geprogrammeerd worden zodat een vooraf vastgelegde opmaak exact wordt gereproduceerd.
- Het kan worden gebruikt voor publicaties in niet-Westeuropese schriften of met wiskundige formules.
- TEX werkt op elk type computer hetzelfde. Het is een *de facto* standaard en het is beschreven in een boek dat in iedere grote boekhandel verkrijgbaar is.
- Door het systeem zo in te richten dat het de mogelijkheid biedt tot structuurbeschrijvende 'markup', kunnen auteurs zich volledig toeleggen op de inhoud van hun publicaties en kunnen ze publicaties meer dan een keer, met verschillende opmaak gebruiken.
- Omdat de opmaak-instructies die in TEX-documenten worden gebruikt in letters worden uitgeschreven is het mogelijk volstrekt probleemloos dergelijke documenten uit te wisselen tussen computers over de hele wereld en dus om publicaties of delen daarvan op te sturen aan vakgenoten en zelfs aan uitgevers.

Hoewel TEX in de academische wereld reeds lang ingeburgerd is, moet het daarbuiten als een belofte voor de toekomst gezien worden. TEX is echter door zijn programmeerbaarheid zowel rechtstreeks, dat wil zeggen als opmaakprogramma, als indirect, als uitvoercomponent van een groter systeem, te gebruiken. De belofte is dus groot.