ELECTRONIC PUBLISHING, VOL. 7(1), 3–19 (MARCH 1994)

TEX in an industrial environment

REINHARD WONNEBERGER

EDS Electronic Data Systems (Deutschland) GmbH
D-65424 Rüsselsheim
Germany

SUMMARY

During its first decade, TEX has been at home mainly in the academic world. Therefore it comes as a surprise to find that it has been spreading into industry during the last few years, and we try to outline some highlights of this development first. Then criteria for an industrial environment application area and reasons for using the structured document processing approach are discussed. It is shown what rôle TEX can play in an integrated document processing environment, and this rôle is exemplified by a case study from application at EDS.

KEY WORDS TEX LATEX SGML Computer services industry Structured document processing

1 THE USERS OF TEX

In spite of its academic image, the use of TEX [1–6] has continually crept into other areas within the past few years. In its column of 'Sponsoring institutions', *TUGboat*, the official journal and newsletter of the *TEX Users Group* (TUG), regularly lists quite a number of large and well-known companies.[1] Even if a company is not in the membership list, there may be some employees somewhere who have managed to get a TEX installation for their own working purposes. Such 'non-academic' usage seems to fall into three categories.

Commercial usage, in our opinion, means that the product of the company is produced with the aid of TEX.[2] This is mainly true for scientific publishing houses such as *Elsevier*, *Springer*, or the *American Mathematical Society*. And because typesetting is connected to their product, they have a completely different attitude toward the complications that are brought about with TEX.

By the term *industrial*, which we will also use to classify our site, we want to illustrate that TEX is mainly used for documentation purposes. In practice, this also means that the fine points of typesetting will only be appreciated if a fast and easy approach to the documentation task is guaranteed.

A third group among profit-oriented organizations in the TEX area might be called *suppliers*. These are either non-commercial institutions that offer public domain installation

[1] Among them are *General Motors Research Laboratories* and *Hughes Aircraft Company, Space Communications Division*, both of which belong, like EDS, to the *General Motors* family, but also distinguished suppliers to EDS, such as *Digital Equipment Corporation* and IBM *Corporation, Scientific Center*. More companies will show up when one goes through the membership list.

[2] While David Ness from *TV Guide* speaks of a 'commercial environment' [7], we talk about an 'industrial environment'.

CCC 0894–3982/94/010003–17

Received 28 November 1991
Revised 7 August 1992

material, or commercial companies that make a living from offering TEX installations as well as TEX-related products or services.[3]

It is interesting to note that, in spite of the highly competitive environment of desktop publishing and typesetting equipment, TEX is now gaining momentum in industry. Perhaps the most prominent example is the North American company *TV Guide*, producing television program booklets in a handy paperback format.[4] This company is also a good example for the 'give-and-take' philosophy of the TEX community.[5] But TEX has also attracted attention in the world of computer manufacturers, e.g. IBM.[6]

There are also examples of software documentation with TEX,[7] including special requirements such as syntax diagrams [12,13], trees [14], and flow charts [15]. In some cases, TEX is used as a postprocessor to other programs, e.g. to print the output of algebraic formula manipulation systems.[8] There is also a 'hidden' use of TEX, either because TEX code is integrated into some other products,[9] or because input is produced in a simplified way by some mail systems (see 'Office applications' below).

2 CHARACTERISTICS OF AN INDUSTRIAL ENVIRONMENT

An industrial environment like ours may be characterized in a few key areas:

- *Group-work* is important in many company efforts, e.g. the production of software documentation. This is quite different from the more common TEX situation of one author writing just one paper or book. As a consequence of group work, there are lots of datasets to be maintained and coordinated.
- The *time-schedule* seems to be always shorter than anticipated, and hours spent on documentation have to be booked and justified.
- *Profit and loss* evaluation seems to be clear only as far as expense is concerned. Quite often, documentation is only seen as a cost factor. The money spent on equipment, computer time, and the like is clearly shown in the balance, whereas it is much more difficult to get corresponding figures for the benefits of writing something down.
- *Large mainframe environments* make documentation a highly complex task and will need a lot of integration work. Let us take the present author's site, Rüsselsheim Information Processing Center (RUIPC), as an example. Although it is by no means the largest IPC within EDS, it is probably the most complex one, supporting both

[3] In the meantime, there is quite a number of such companies, as can be seen from the advertisements in *TUGboat*.

[4] The figures for this company are quite impressive; annually they produce nearly *one billion* copies of the magazine and about *one million* different pages bound in *five thousand* different magazines, *two hundred* pages each. An overview of their experience was presented by David Ness at the 1987 TUG conference [7]; the more technical problems of typesetting are presented in [8].

[5] David Ness has by no means been the only one to illustrate the 'give-and-take' philosophy of the TEX community by having served until recently as a board member of the *TEX Users Group*.

[6] Inside IBM, TEX is used by several individuals and groups on there own initiative, and there are drivers for several IBM printers; e.g. [9] which was produced with TEX on an IBM 4250 electro-erosion printer. To help customers without access to the networks or the TEX user groups, it is even available as a program offering (PRPQ) for AIX, IBM's version of UNIX; see [10].

[7] To give just one example, the several hundred pages of documentation for the ABAQUS product are apparently typeset with TEX [11].

[8] E.g. REDUCE [16], MACSYMA [17], or MAPLE [18].

[9] E.g. VAX DOCUMENT from DEC [19], or dbPublisher [20].

engineering and commercial services,[10] and the variety of the services offered is also quite diverse.[11]

2.1 Alternative concepts

For many people it is by no means obvious why one would like to install TEX in an industrial environment. Printing requirements have been fulfilled over the years with different tools, and so there must be a good reason to bring in yet another tool.

The most popular typewriting tools seem to be PC programs that allow the manipulation of the text directly on the screen, working according to the WYSIWYG acronym, which stands for: 'What you see is what you get.' There are two main reasons why these tools are widespread. A PC, like a horse or a car, is one's own. And producing text with it is similar to typewriting, only much more powerful.

The idea of 'My PC is my castle' can still be kept with a PC installation of TEX. However, the markup approach that is normally used with TEX, especially if LATEX [21] is used, requires a totally different approach to typesetting. Public opinion in the area of typesetting is preoccupied with the battle of the different commercial products, but despite its desultory coverage in the early days,[12] today TEX is more accessible to a wide audience.[13]

On the other hand, TEX is by no means the first tool to offer a general mark-up approach.[14] Proliferation in industry, however, seems to have been effected mainly through IBM's DCF, formerly known as SCRIPT/VS, and its academic twin version WATERLOO SCRIPT.[15] Thus, in a traditional mainframe environment, TEX has to compete with DCF, and typographic habits seem to be in favour of DCF.[16]

Thus it should come as no surprise that DCF is a *de facto* standard in the world of large mainframes, both in terms of a software tool and of application and layout habits. People having that background will inevitably ask why one should bother to introduce yet another document processing system. The main arguments for TEX seem to fall into two categories, functionality and environment. Let us first deal with functionality.

[10] In addition to the traditional MVS platform, there are also six other platforms, namely UNICOS (Cray); VMS (DEC VAX); HP-UX (HP); DIPOS (Nixdorf); MPE/XL (Bull).

[11] Engineering: multi customer Cray-service, vector computing, models and methods (finite elements), graphics, esp. CAD, CAM, CAE; Commercial applications: large database applications (IMS, DB2), transaction processing (CICS), complex batch processing and scheduling, large range of individual data processing applications; Systems integration: operation of six different environments, support for special systems, systems integration; Communication: SNA and non-SNA, X.25 (DCS, SNA-PAD), LAN (Ethernet); Document processing: integrated document processing, high-quality volume printing, demand and distributed printing.

[12] E.g. Furuta & MacKay, *Dr Dobb's Journal*, 1985; Whitby, *DTP Desktop Publishing*, 1987; *Computer News*, 1987; Brown, *The Scientist*, 1988; an example in German is Schelhowe, *Computer Magazin*, 1989. See also the moderate but continuous amount of advertising by commercial TEX suppliers, by which TEX is positioned as a specialist's tool for typesetting math.

[13] See the *Chip Special* on LATEX [22], which was also typeset with LATEX, and the public domain offer of OzTEX (for Apple) in *Macwelt* 7/92.

[14] For a general overview, see [23]; LATEX has been influenced by SCRIBE [24].

[15] Virtually all technical documentation at IBM is produced with this tool, making IBM one of the largest publishing houses worldwide. In addition, a lot of documentation is produced this way by IBM customers.

[16] On the one hand, many users want their documents to look like the IBM manuals in layout, and, on the other hand, it is not an easy task to write a corresponding style in TEX. It should also be noted that DCF comes with a licensed version of Times Roman, which, having been optimized for the rather small resolution standard of 240 dpi (dots per inch) used in IBM production printing, gives much better results than the Computer Modern fonts, which, in general, only look good at a high resolution. This problem has been eliminated by the New Font Selection Scheme [25] and a font conversion program, allowing the use of the DCF fonts under TEX.

- *Quality of typesetting*. Although recent versions of DCF do provide good quality as far as in-house publishing is concerned, there are some internal limitations due to upward compatibility and the typesetting philosophy underlying DCF. TEX, by default, provides a quality conforming to the highest standards of professional typesetting, at no additional cost.[17]
- *Typesetting formulas*. Math Typesetting in DCF requires the Script Mathematical Formula Formatter (SMFF) [26] to be installed; in an industrial field, it may be difficult to justify the additional cost, if only a few formulas are to be typeset, and in fact most installations that I know about do not have this product. In TEX, math typesetting is an intrinsic part of the program, and there are quite a few macro packages that allow the typesetting of other types of formulas, such as chemical structure formulae [27], trees and the like [14].
- *Diagrams*. Though both programs can include graphics, only LATEX offers to produce diagrams as part of the source. In spite of some limitations, this concept is well-suited to cover everyday needs, e.g. to show computer configurations or display data as a business graphic.[18]
- *Bibliographies*. Creating bibliographies is a tedious and error-prone task which has been improved to a large extent by BIBTEX.[19] There is no official answer to this requirement in the world of DCF, and having no easy way to manipulate bibliographic data can only mean that existing and costly information cannot be exploited at its best.

Whether these items of functionality are important or not will depend on the specific requirements. As investment in documentation is higher than most people imagine, the following topics may be more important, at least in the long run.

- *Cost*. DCF and its companion products cost a considerable amount in license fees. TEX may be obtained for free, especially so for mainframes.
- *Vendor dependence*. Building on DCF implies dependence on IBM's policies and prevents a switch to non-compatible operating systems and hardware.
- *Standardization*. Whereas DCF builds on EBCDIC sources and produces printer-specific output, TEX is standardized on both sides: input is in ASCII, and output is in the device-independent `.dvi` format and can be printed or viewed at any other installation worldwide, provided no unavailable fonts are used.
- *Portability*. One major drawback of DCF is that its use is confined to large IBM mainframe installations.[20] TEX is available on any operating system from notebook to supercomputer. But there are also several organizational consequences. In many firms, mainframe-oriented people will use DCF, whereas business-oriented people will prefer PCs and workstations, using WYSIWYG-type wordprocessors on them. This inevitably prevents a free flow of documentation in source format and close cooperation on common documents. Although standards like ODA (office document

[17] This topic would need an in-depth discussion of typesetting algorithms, which would go beyond the scope of the present paper.

[18] Creating diagrams is now supported by the mouse-driven PC-program TEXcad [28].

[19] BIBTEX. will automatically create a bibliography from a bibliographic database, converting entries to a specific bibliographic style (for details see [29]).

[20] Though some efforts were made to extend it into the PC world, it is still unclear whether they will be accepted by the users.

architecture) or SGML (Standard Generalized Markup Language) may help with respect to sources, TEX, as far as we can see, is still the only way to span all processing platforms, which will even become more important with the continuing growth of the workstation share.

- *Stability*. Errors to affect the ordinary user have been very rare, and in the history of TEX there has been only one release change, which was mainly an upgrade from 7-bit to 8-bit ASCII. Now TEX is virtually frozen.[21] In this way, TEX has gained a degree of stability unheard-of in the industry.
- *Customization*. By being open and well-documented, TEX offers the unique opportunity of being used and customized in connection with other programs, an advantage that is especially important for computer companies.[22]

2.2 The structured markup approach

A very important point in the industrial environment, and the common denominator shared with DCF and SGML [33–35], is to use the 'structured document' approach [36,37].

On June 27, 1988, a train crashed into the gare de Lyon *in Paris at high speed and without braking. More than fifty people were killed in the accident and a considerable number wounded. The accident was found to be caused by the coincidence of several bad circumstances, one of which was closely related to erroneous typography. The crucial piece from the instruction paper is shown here, cf. the facsimile in [38] Fig. 1-a, p. 22.*

Plusieursvehiculessontbloques,lemecanicien:
S'assurequeceblocagen'estpaslaconsequencedelafermeture
d'unrobinetd'arretdelaconduitegeneralesitueavantlapartie
detrainbloquee:
Actionnerlerobinetd'arretCGn'estferme:
Ilactionnelacommandedelavalvedepurgeletempssuf-
fisantpourprovoquerledesseragesurchaquevehicule
bloque.
Pr(Si plusieurs)robinetd'arretesttrouveferme:
Ilouvrelerobinet
⇒Dansles2cas,lemecanicien:
ouvrelerobinetd'arretCGsitueenarrieredudernier
vehiculereliealaCG.
verifieleserragedesfreinsduderniervehiculefreine.
refermelerobinetCG
verifieensedirigeantverslatetedutrain:
ledesseragedesfreinsdetouslesvehicules.
quelablocagen'apasprovoqued'avarieauxroues.
Ilappliquelesmesuresconcernantlesignalementetlareprisedemar-
che(article385).

The basic problem comes after the second item in the list, which is marked here by an arrow (⇒). Whereas the text clearly states that the following actions be applied to both *cases, typographic stucture implies that these instructions are only a part of the second case. So it is very likely that the reader will skip these actions if case 1 applies.*

Figure 1. P.G.M. Chapitre VI, Article 316 b

[21] Don Knuth has decided to discontinue the normal practice of upgrading from 3.1 to 3.2 etc.; instead, version numbers will converge to π for TEX and to e for METAFONT. This is Knuth's (mathematical) way of expressing that correction of true errors is still possible, but functionality of the programs is frozen definitely [30]. Further development [31] is still possible, but will have to happen under a different program name [32].

[22] See the examples in footnote 9.

As is shown by the example in Figure 1, non-structured documentation may lead to disastrous consequences. Although this example is certainly not an ordinary case, texts in an industrial environment are very often *directives* in the sense we just saw. They induce or guide action which may have severe consequences if misunderstood. In addition, texts in industry are very often directed to an environment where there is not enough time or concentration for accurate reading.[23] The most important piece of information should be found instantaneously without first plunging through pages of context.

Hence typographic structure should be derived automatically from the internal stucture of a document, which in turn should be specified explicitly by the author. In addition, the concept of *generalized markup* helps authors to produce good documentation and assures consistency between different documents. In order to achieve these effects, we use LaTeX as the only official interface to TeX.

2.3 Reasons for TeX

In a company like EDS documentation is produced by both PC- and host-based systems. In a comparison of these two areas, it becomes clear at once where one of the major benefits of installing TeX lies. TeX is at the moment the only system that can reconcile all our different platforms under the common roof of one document processing language. This integrational aspect is extremely important in a company like EDS, which is engaged in systems integration. When new customer sites are brought in on a facility management contract, the portability of TeX will allow them to share our documentation facilities almost immediately. Of course, there are also several other reasons for introducing TeX, as is shown in Figure 2, one of which is the educational benefit in using TeX, as a growing number of new employees will bring expertise in TeX with them from their student days.

1. Portability
2. Device-independence
3. Wide-range consistency
4. Integration of PC and mainframe worlds
5. Full access to code
6. Full access to documentation
7. Coverage of many different business areas
8. Long-range stability
9. Low costs
10. Innovative user group
11. Education benefit (students)
12. Literate programming [39]

Figure 2. Reasons for TeX

3 INTEGRATED DOCUMENT PROCESSING

For us, installing TeX means to make it part of our *integrated document processing* concept (IDP).

The key points of the IDP concept are shown in Figure 3. This concept allows us to take full advantage of the benefits of our mainframe environment, such as data security *(access control system)*, data protection and availability *(backup)*, data sharing and archiving *(projects and proposals)*, consistency and corporate design, centralized software and support, resource sharing *(high-end printing equipment)*, postprocessing *(binding)* and distribution.

[23] This 'non-usage' of text is very well reflected in the ironic slogan: *If everything fails, read the instructions.*

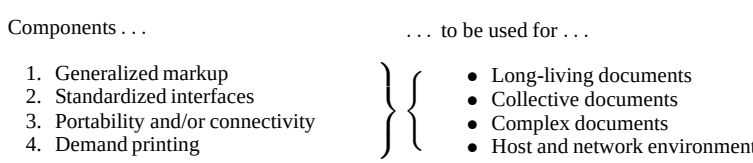

Figure 3. The IDP*-concept*

However, there can be no doubt that some extra effort is needed to make an academic-born system like TEX fit into an industrial environment.

Normally, software is bought from vendors, and considerable money is paid for maintenance. With TEX, software does not cost money in this sense. Although errors are corrected, there is no maintenance in the classical sense. This clearly indicates that installations have to develop at least a moderate know-how of their own if they want to succeed, especially in the area of the MVS operating systems. On the other hand, as everything in the software is accessible and well-documented, every installation is in a position to customize TEX according to its own needs and to implement whatever might be desirable.

Excellent support is provided by the TEX Users Group (TUG) and its regional counterparts.[24] In addition, there is now support for many languages, such as Russian, Polish and others, including even languages with a different writing technique such as Japanese, Arabic, or Hebrew.

There is a growing conception that some of the non-profit organizations deliver top-quality service, as was elaborated in a recent article in the well-known *Harvard Business Review* [41]. The main point seems to be that non-profit organizations are not hampered by business restrictions in pursuing their goals.[25]

Thus, a major difference is that TEX development is function-driven, whereas commercial competitors are market-driven. The main goal of TEX is to meet typographic challenges, whereas the main goal of the competitors is to produce profit for their companies.

Much of TEX development and communication is done via worldwide networks. In the USA, it is quite common for development personnel in industry to be on the networks. Even in Europe, where there has been a traditional segregation between universities and research institutions on the one hand and the industry on the other, cooperation is increasing. Thus, TEX is truly international, complying with the general trend of software *globalization*.[26]

In an industrial environment, it would not make sense to bring in TEX with a short-term perspective. As experience shows, some of the programs or procedures to be documented may last for about twenty to thirty years. To be convinced of a long-lasting future for TEX,

[24] DANTE for speakers of German, GUTENBERG for French, NTG for Dutch, and UKTEXUG for English in the Old World, and also the growing number of more or less formal groups in the countries of Eastern Europe and other parts of the world. These groups provide software distribution, archives, discussion lists, meetings and conferences, education, consulting, and a platform for individual communication. Full details are given in the 'TEX Users Group Resource Directory' [40]. TUG will provide a free parcel of starter information on request. The address of TUG is: TEX Users Group, PO Box 869, Santa Barbara, CA 93102, USA; Telephone: +1 805 963 1338; Fax: +1 805 963 8358; Internet: `tug@tug.org`.

[25] A good example for that claim is the case of the TEX user community at large, where new software is often distributed free of charge.

[26] We borrow this term from Anthony C. Hearn, the author of *Reduce*, now dealing with such more general issues as an analyst at the Rand Corporation.

one needs only to take into account the huge amount of documentation which lies hidden behind the set of TEX installations worldwide.[27]

For our documentation tasks, we clearly favour the generalized markup approach, which had already been established in our use of DCF. So it was quite natural that we decided to offer only LATEX as the user interface to TEX. This decision has helped us in several ways. LATEX documentation is much shorter to read, e.g. [42], and much easier to understand, e.g. [43,44], than TEX documentation.[28] LATEX offers much more document support, such as all sorts of automatic countings, cross-referencing, and especially the famous tabular services. For someone who is already familiar with DCF/GML, it is quite easy to switch over to LATEX, which differs mainly in syntax.

Integration from the user's point of view means that TEX is part of our normal TSO/ISPF working environment. Rather than an ISPF menu approach [47], we chose an editor-driven approach under the general roof of our USE concept, USE standing for *universal source environment*. We have also managed to provide some editing support, which can spare the user much trouble in typing LATEX environments and things like that; see [48]. Although it takes some extra thinking and effort to adapt TEX to the MVS-world,[29] the result has proved to be a powerful and comfortable working tool, incorporating such useful features as Partial Execution, which will save user and machine time as well as printing cost.

In addition to LATEX, we offer our users the full range of TEX-related products, among them: BIBTEX, *MakeIndex*, METAFONT and the auxiliary programs. This service is also important in connection with other services such as *Reduce*, the well-known formula manipulation product [16].[30]

From a printing point of view, TEX is fully integrated into our AFP-based [49] integrated print services (IPS) concept. Our print command IPS will automatically convert the device-independent .dvi format produced by TEX into .list3820 files, either to be kept in a dataset for later printing on demand, or to print it by PSF on a page printer.[31] The dvi printer driver was modified to support AFP *copygroups*,[32] and to pass extensive comments into the `.list3820`-file, which are quite useful for determining the status of these files later on.

AFP print resources are also shared by our peer installations like the one on VAX–VMS, which only needs to upload a job with the dvi file, allowing the VAX users to profit from the benefits of IPS printing. On systems without a peer installation, e.g. Cray–Unicos, users can send their input to the MVS system to be run through TEX, BIBTEX, and *MakeIndex*, and to be printed automatically. Transcript files such as `.log` are routed back on request. In fact, the whole production cycle can be run in batch, allowing remote execution from any node in the worldwide EDS network.

[27] A good guess may be more than 3500 mainframe and an unknown number of PC installations, which may go well beyond 1 million.

[28] The original book by Knuth is for experts only. There seems to be a growing demand for simplified and excerpt-like TEX-books, e.g. [45], [46].

[29] Unbelievable as it may sound, one of the most troublesome points is handling *square brackets*, because they virtually do not exist in the IBM 3270 world.

[30] For this product, we can provide a bibliography in BIBTEX format, which is maintained and distributed over the electronic networks. Thus, users of Reduce can not only find interesting publications, but also quote them with all the accuracy, style adaptability and comfort offered by BIBTEX.

[31] Either on an IBM 3820 family (240 dpi) or on an OEM printer emulating AFP, or by the IDA emulation program [50] on desktop printers (300 dpi), which in many cases also serve as printers for PCs simultaneously.

[32] This makes possible techniques such as printing electronic forms (overlays) or drawing paper from different bins.

In addition, we can also convert `.list3820` pages into AFP overlays [51], which are then used in production printing. This is an efficient way to print direct mail or newsletters, the addresses being printed into the envelope window field as simple line data.

The full benefits of TEX in a large machine environment like ours can only be realized when one can manage to run a peer installation setup [52]. In our environment, the TEX program is run on several different machines under a growing number of operating systems.[33]

If TEX is to be used not only by mainframe people there will always be the question: 'Can it be used by my secretary?' According to our experience the answer depends on motivation and education. We have several secretaries within EDS who can use DCF, and some of them also use LATEX,[34] but they work in an environment where it is easy to get help from a colleague. Things might be a little bit more difficult, however, if the boss has no idea about document processing and PCs, and the secretary is all on her or his own.

So, using TEX in industry is not so much a question of a single person but of the culture of a whole group or even company.

4 USAGE AT EDS

In car manufacturing, commercial applications such as book-keeping on the one hand and engineering applications such as computer-aided design on the other, have been traditionally processed in different computer environments. EDS has stepped in to integrate these environments, and this also holds true for our document processing. An overview of our present IDP applications is given in Figure 4, the share of LATEX depending on the specific requirements in each field.

1. Production Printing
   (a) Newsletters
   (b) Direct Mail
   (c) Forms and Overlays
2. Technical Documentation
   (a) Program Description
   (b) Program Development
   (c) Scientific or Technical Contributions
   (d) Standards Sheets
   (e) Bibliographies
3. General and Office Documentation
   (a) Proposals
   (b) Planning Concepts and Schedules
   (c) User Manuals
   (d) Presentations
   (e) Meeting Minutes
   (f) Letters and Memos
4. Special Documentation
   (a) APL
   (b) Diagrams and Formulae

Figure 4. Application areas

Engineering documentation It is quite natural that TEX is used more heavily in engineering than in commercial surroundings. Many engineering applications involve mathematics, and here is the natural domain of TEX; see [53].

[33] At present MVS, Unix, VAX–VMS, MS–DOS, MAC–OS, Atari–TOS. Owing to our decentralized approach in document processing, it is left to the different user groups to upgrade their macro installations according to their need. This can be done by downloading macros from the MVS lead installation, which is officially maintained.

[34] It may be interesting to note that the first secretary to learn LATEX on our installation, Sylke Thies, did so mainly on her own, guided only by the *Kompaktführer* LATEX [42].

Procedures provided by our CSE group allow conversion for all relevant paths for

- Applications,
- Graphic Systems and
- Printers / Plotters,

using the GDF Metafile.

Application	Graphic Syst.	Output Device
Radioss	Cadam	IBM 5080
Abaqus	Catia	IBM 5084
Nastran	CGS	IBM 3279
Star	Caeds	IBM 3193
GDQF	Nastplot	IBM AFP Printer
ICU	GDDM	IBM 4250 Erosion Pr.
TEX, DCF	SAS	Colour Graphic Needle
Fortran	Dissplа	Colour Raster Plotter
PL/1	PHIGS	Pen Plotter

Figure 5. Graphic Filters

Graphics inclusion Through our AFP interface, we are in a good position to include most types of graphics both in DCF and TEX documents. Figure 5 gives a rough idea about the packages involved.

Production printing Through the conversion process mentioned above, we can fully integrate TEX output into our AFP environment. This is very useful for direct mail printing, as ready-made page overlays are combined with the few print lines that are necessary for the address. This technique, which we also use for our *Quality Newsletter*, also allows us to combine pages from different sources, e.g. DCF and TEX.[35] The printing process may also be handled through our production output dispatching system, allowing us to produce output bundled according to recipients and simultaneous printout at different locations.

Corporate publishing As mentioned above, we also produce the in-house newsletters with LATEX. To conform to our corporate design, they are printed in three columns in a firm grid and with a Times-like typeface.[36]

Another benefit is the possibility of giving TEX-produced documents (`.dvi` files) or sources to professional typesetters for high resolution typesetting,[37] In addition to excellent typesetting quality, this method ensures accuracy, as the error-prone re-typing is eliminated; it saves cost through in-house preparation and speeds up production.

Documentation and publication One of our most important applications for TEX is software documentation, where it is important to distinguish between commands, variables, and explanation, and to give diagrams every now and then.

But TEX also qualifies to fulfil the requirements posed for documentation of our Quality Management System by the ISO 9000 series of quality management standards.

[35] In combination with METAFONT, this concept gives us excellent control of fonts. It should be noted in passing that METAFONT, in addition to being a powerful font manipulation tool, can also serve for other useful purposes, e.g. to produce logical symbols for electronic components or to help with the typesetting of images; see [10]. Thumbing through *TUGboat* [54] will show further applications.

[36] We created a specific document style which relies on the `multicol.sty` [55] and the new font selection scheme (NFSS) [56,25]. After solving a few problems with the TEX font metric (`.tfm`) information, codepages, and *umlauts*, we can now use the original AFP fonts. These fonts are the DCF functional equivalents of Monotype Times New Roman (with serifs) and Monotype Arial (without serifs), and they were optimized manually to work well at the AFP resolution of 240 dpi [57–59].

[37] This is shown by [42], which was typeset in Times and Helvetica at 1000 dpi.

Office applications The one thing especially popular in office applications is the enhanced LATEX tabular environment [60,61]. It is used to produce quite a number of overviews and reports, many of them in landscape format.

Another important application is to achieve fine print from our general-purpose mail system, which is available to the vast majority of employees; see also [52]. For standardized documentation, it may be useful to produce a reduced type of input with the aid of masks. Quite a lot of similar applications are described in [62].

Special applications In some cases, TEX is also used as a sort of database or general system. For example, many of our managers have chosen to prepare their budget with the aid of a special LATEX tabular environment allowing a kind of spreadsheet calculations.[38]

The document processing aspect is also stressed by some of our groups who use it for internal work documentation by tricky macro definitions: they manage to produce different kinds of reports from one and the same set of data, thus emulating functions of a database system which would be much more complicated to handle otherwise.

A special application XDOC was developed by Klaus Schmidt to allow automatic generation of program source code, user documentation, programmer's guide, and help files from one and the same XDOC source. As all documentation is a comment with respect to the used programming language, sources can be compiled without XDOC processing.[39] As this approach is rather general, several programming languages are supported.

Bibliographic applications When we started to work with BIBTEX, we soon discovered that it is an excellent tool to help us manage the large number of internal documentation papers and brochures.[40]

Education LATEX is also used in our education department to write course material.[41]

For education purposes, it is also quite helpful that manuals and slides can share parts of the input. A special style allows printing handouts of the presentation slides in a two-column layout, slides appearing as column-wide figures.

Outside relations Though most of our documentation with TEX is done for internal purposes, it has also been used in several cases to produce proposals for customers and papers submitted to international conferences, not to mention here the contributions to TEX conferences.

Conclusion To sum up, TEX helps us develop our IDP concept since it improves the appearance and internal quality of documents, the skills of the users, the cooperation in the company, and the productivity, allowing us to live up to one of the basic beliefs of EDS: "We are committed to excellence" — excellence in typography and document processing being an important incentive for other aspects of work and service.

[38] The main advantage over normal spreadsheet programs is that it can be used from any terminal and allows inclusion of footnotes to explain budget changes and use all possibilities of typesetting at the same time, e.g. headlines, diagrams etc.

[39] This approach is also used in the DOC option for macro files [63], but is one of the main differences to the literate programming [64,39] approach in WEB [65,66], which requires the code to be run through a program called TANGLE to produce compiler input.

[40] Our work with BIBTEX also showed that some additional features are desirable [29].

[41] The first course in LATEX was written by Johann Dentinger in early 1988, giving an overview on available software tools, and it makes heavy use of *tabulars* and *pictures*. In the meantime, a few more brochures were produced that way, covering topics such as JCL, TSO, SLC, MVS, PACBASE, FOCUS, COBOL. Of course, there are also many internal papers, describing software, hardware, or operational procedures.

5 REQUIREMENTS

Of course, light goes with shadow, and so there are a few weak points in TEX, that hopefully will be improved in the near future.

International language support is not very elaborate. The current version of TEX[42] has removed some of the nasty obstacles in everyday life with TEX, but is not yet the functional upgrade one might wish, and features such as a spelling check are not even considered.

In some cases, it is desirable to trigger system processes from within TEX.[43] An application for this is to call BIBTEX and *MakeIndex* to make a new bibliography and index within TEX and then typeset bibliography and index, thus saving one pass through the whole manuscript and adding user comfort. Other applications include floating-point arithmetic or algebraic calculation; remember also the spreadsheet application mentioned earlier. The system command might also serve as a hook for external hyphenation.[44] In present TEX, there is no official way to use features of the operating system from within TEX. It seems, however, that this problem can be solved in the near future.[45]

Explicit typographical typesetting is rather difficult and sometimes almost impossible, e.g. letting text flow around irregular figures, and rotation is not provided inside TEX.[46] The topic of rotation, either of whole pages, or of fields within a page, plays a much more important rôle in industrial documentation than in books.

Whereas these requirements do not touch the true spirit of TEX, there are some others that are more debatable. One of them is PostScript, the other is colour support.[47]

As far as LATEX is concerned, many desirable designs can be implemented in the current framework of LATEX.[48] Many more useful features of that kind can be added, and some of them already exist, waiting only for the final touch and official inclusion into some distribution. Some other features, however, need fundamental changes within LATEX or even within TEX.[49]

For LATEX, an attribute concept is highly desirable:[50] DCF/GML and SGML tags allow named attributes, which are rather similar to the fields in BIBTEX. Though this concept is not necessary for normal typesetting, it is quite useful for more specific applications such

[42] TEX 3.0, which was announced by Don Knuth at the 1989 TUG meeting celebrating the tenth anniversary of TEX, provides some basic features to improve this situation: (1) 8-bit character support in all WEBware, i.e. all programs of the TEX system, (2) multiple hyphenation tables, (3) extended hyphenation and ligature support.

[43] From a TEX point of view, such a system command should behave like the `read` and `write` primitives. Additionally, a return code should be passed back to TEX.

[44] Special care should be taken to preserve as much portability as possible also for these features.

[45] The system command written for MVS by Ferdinand Hommes from Gesellschaft für Mathematik und Datenverarbeitung (GMD) is a TEX modification confined to one operating system, and source files using it will no longer run on other installations. A general solution to this problem might be to use a non-existing input and output stream (e.g. '99') for communication with the system. As Don Knuth said recently in private communication, he would not consider such an implementation as a violation of his decision to keep TEX frozen (cf. footnote 21), thus making the way free to add this important feature to all TEX implementations in a standardized way.

[46] It may by achieved if supported by the DVI driver.

[47] In this context it may be important to notice that PostScript is supported in DCF version 3.2 and colour separation support with version 4.0.

[48] E.g. the `ftnright.sty` [67], which is also exemplified in [29,31,68].

[49] For the current state and history of the LATEX 3 project, see [69].

[50] A straightforward attribute concept for TEX was presented by Wolfgang Appelt in [70]5.3, pp. 58–63. It is based on TEX's parameter scanning mechanism, but has the disadvantage of introducing new syntax. So it might be worthwhile looking for a concept more along the lines of LATEX, or even breach the iron law of upward compatibility and use an SGML-like concept.

as keying in addresses or other database-like information. In addition, it plays a key rôle in SGML.[51]

Audacious as it may seem, the best investment in an industrial future of TEX might be to implement an SGML-like syntax for the new release of LATEX,[52] or to write an SGML interpreter in TEX.[53]

A new challenge is the growing trend to distribute documentation on electronic media, which in turn requires a hypertext approach.[54]

The most difficult problem seems the organization of future support and development. Both TEX and LATEX owe their current state of quality to the devotion of the two individual wizards Don Knuth and Leslie Lamport, both of whom have resigned from being the focal points for their products, pointing out that every line of code can be changed by any user if the result is not called TEX, or LATEX, thereafter.

On the other hand, much of the strength and value of the TEX products results from this strict standardization. If different users start to program similar modifications, then this can only lead to chaos. As no one can deny the authors the right to deal with other themes, the problem can only be solved by a strong coordinating authority. The TEX Users Groups in a formal sense and the TEX community in an informal sense are doing their best to step in and make every effort to adjust themselves to the new challenge.[55] All TEX users interested in evolution should specify precise requirements, following the excellent example of [79] which triggered a new version of TEX (3.0).

All in all, our experience at EDS can encourage both the TEX community and industry to profit from each others' virtues and specific experience.

ACKNOWLEDGEMENTS

I would like to thank Karen Brühl, Dr Hans-Jürgen Lustig, Frank Mittelbach, Sylke Thies and the anonymous referees for their help in producing this article. Frank Mittelbach has also had, in addition to his own published contributions, a key rôle in implementing some of the concepts described in this article.

POST SCRIPTUM, AUGUST 1994

Since the files were closed on this paper (and also on its bibliography) in early 1992, the 'river TEX' has slightly modified its direction and flow.

The growing influence of PostScript as already known from the dvips family of printer drivers also led to MetaPost, a MetaFont with PostScript output. TEXware is now distributed

[51] This topic was brought up by Malcolm Clark at an early time in TEXline [71]; see also [19,72,73].

[52] This idea was outlined in [74]; see also [75].

[53] A prototype of such an SGML interpreter written by Frank Mittelbach may qualify to handle our Quality Management System documentation. Although it does not check document structure as strictly as a true SGML parser, it allows interpretation of DTDs, with some minor restrictions. Being based on LATEX, existing LATEX macros can be used in a very simple and efficient way to implement SGML tags. The benefit of saving another layer of software outweighs by far the deficiencies with respect to a fully fledged SGML implementation.

[54] For this approach in connection with TEX, see [76,77].

[55] A major step in the right direction was made at the 1989 TUG meeting, where it was decided that TUG will become more international than before, and that the chairpersons of the early European TUG groups will be vice-presidents of TUG. In addition, TUG is sponsoring some development work and hopefully will find ways, for both TEX and LATEX, to coordinate ongoing development work on the two products and customize this work into standardized releases and distribution packages; see also [78].

on Internet by the CTAN Comprehensive TEX Archive Network and will soon be available on CD-ROM.

Perhaps even more importantly, LATEX 2_ϵ, the new intermediate version, and the full rewrite to follow as LATEX 3, are going to change the character of LATEX from a self-contained to a *kernel* system, supporting enhancements like the NFSS New Font Selection Scheme and inviting the user to choose from the large collection of supplementary style files and packages as described in *The* LATEX *Companion* (by Goossens, Mittelbach and Samarin; Addison-Wesley, 1994, ISBN 0-201-54199-8).

In our industry, the last two years has seen tremendous change, forcing computer giants on to their knees and leading to a general shake-out of anything that does not immediately produce business results. Although EDS was not affected directly, TEX support at our site was reduced to an absolute minimum, only to see a need for it arise again via another route, caused by a commercial package which generates LATEX to print invoices for cellular phone customers.

The growing consciousness for quality caused us to embrace the ISO 9000 standard of Quality Management, and in order to produce the corresponding documentation in a standardized way, we studied SGML, and Frank Mittelbach wrote a prototype style package for LATEX that, to a certain extent, reads Document Type Definitions (DTDs), translates them into LATEX commands, and this allows immediate typesetting of such documents. In our view, all efforst should be taken to bring SGML and LATEX together as close as possible.

And we are quite curious to see what changes will be brought about during the next two years.

REFERENCES

1. Donald E. Knuth, *Computers and Typesetting*, Addison-Wesley, Reading, MA, 1986. Consists of [2,3,4,5,6].
2. Donald E. Knuth, *The TEXbook*, Volume A, Addison-Wesley, Reading, MA, 1986. Hardcover: ISBN 0-201-13447-0, Softcover: ISBN 0-201-13448-9.
3. Donald E. Knuth, *TEX: The Program*, Volume B, Addison-Wesley, Reading, MA, 1986. ISBN 0-201-13437-3.
4. Donald E. Knuth, *The* METAFONT*Book*, Volume C, Addison-Wesley, Reading, MA, 1986. Hardcover: ISBN 0-201-13445-4, Softcover: ISBN 0-201-13444-6.
5. Donald E. Knuth, METAFONT*: The Program*, Volume D, Addison-Wesley, Reading, MA, 1986. ISBN 0-201-13438-1.
6. Donald E. Knuth, *Computer Modern Typefaces*, Volume E, Addison-Wesley, Reading, MA, 1986. ISBN 0-201-13446-2.
7. David Ness, 'The use of TEX in a commercial environment', in *TUG87*, ed., Dean Guenther, number 5 in *TEXniques. Publications for the TEX Community*, pp. 115–123, P. O. Box 9506 Providence, R. I. 02940, U.S.A., (1987). TEX Users Group.
8. Elisabeth M. Barnhart, 'TEX in the commercial environment. Setting multi-column output', *TUGboat*, **8**(2), 184–189, (1987).
9. Friedrich M. Wahl, *Digitale Bildsignalverarbeitung. Grundlagen, Verfahren, Beispiele*, Springer, Berlin etc., 1st edition, 1984. ISBN 3-540-13586-3, ISBN 0-387-13586-3.
10. Richard O. Simpson, 'Nontraditional uses of METAFONT', in *TEX. Applications, Uses, Methods*, ed., Malcolm Clark, Ellis Horwood Series in Computers and their Applications, pp. 259–271, Chichester, (1990). Ellis Horwood. TEX88. Proceedings, Third European TEX Conference. Exeter, England, August 1988.
11. Hibbitt, Karlsson & Sorenson, Inc., *Abaqus. 1: Theory Manual, 2: User Manual, 3: Example Problems Manual*, 1987.

12. Mary McClure, 'TEX macros for COBOL syntax diagrams', in *TUG89*, ed., Christina Thiele, volume 10,4 of *TUGboat*, pp. 743–750, (1989).
13. Bjørn Kirkerud, *Systematic Object-oriented Programming with Simula*, Addison-Wesley, 1st edition, 1989. ISBN 0-201-17575-6, 544 pages, USD 42.50.
14. Anne Brüggemann-Klein and Derick Wood, 'Drawing trees nicely with TEX', in *TEX. Applications, Uses, Methods*, ed., Malcolm Clark, Ellis Horwood Series in Computers and their Applications, pp. 185–206, Chichester, (1990). Ellis Horwood. Also in *EP-odd* **2**(2).
15. Michael Ramek, 'Flow charts with TEX', *TEXline*, (14), 21–26, (February 1992).
16. G. Rayna, *Reduce. Software for Algebraic Calculation*, Springer, Berlin etc., 1st edition, 1987. ISBN 3-540-96598-X; Preface by A. C. Hearn.
17. J. Ph. Chancelier and A. Sulem, 'MacroTEX: Un générateur de code LATEX implémenté en MACSYMA', *Cahiers Gutenberg*, (3), 32–39, (October 1989).
18. Michael Lavaud, 'AsTEX: An integrated and customizable multi-window environment for scientific research', *Cahiers Gutenberg*, (10–11), 93–116, (September 1991).
19. Alan E. Wittbecker, 'TEX enslaved', in *TUG89*, ed., Christina Thiele, volume 10,4 of *TUGboat*, pp. 603–606, (1989).
20. David Penfold, 'dbPublisher', *TEXline*, (11), 17–18, (September 1990).
21. Leslie Lamport, *LATEX: A Document Peparation System*, Addison-Wesley, Bonn, 1st edition, 1986.
22. Rudolf Potuček and Stefan Schwarz, LATEX — *Satzkunst statt DTP*, CHIP-Anwender-Praxis, Vogel-Verlag, Würzburg, 1st edition, 1992.
23. Richard Furuta, Jeffry Scofield, and Alan Shaw, 'Document formatting systems: Surveys, concepts, and issues', *ACM Computing Surveys*, **14**(3), 417–472, (September 1982).
24. Unilogic, Ltd., *Scribe Document Production System. User Manual*, April 1984. Developed by Brian Reid.
25. Frank Mittelbach and Rainer Schöpf, '[Reprint (with corrections):] The new font family selection: User interface to standard LATEX', *TUGboat*, **11**(2), 297–305, (June 1990).
26. International Business Machines Corporation, *Document Composition Facility. SCRIPT. Mathematical Formula Formatter. User's Guide*, 1st edition, June 1989. S544-3306-00.
27. Michael Ramek, 'Chemical structure formulæ and x/y diagrams with TEX', in *TEX. Applications, Uses, Methods*, ed., Malcolm Clark, Ellis Horwood Series in Computers and their Applications, pp. 227–258, Chichester, (1990). Ellis Horwood. TEX88. Proceedings, Third European TEX Conference. Exeter, England, August 1988.
28. Georg Horn, *TEXcad Version 2.7*, Burgweg 1a, D-5400 Koblenz-Moselweiß, September 1990. Shareware, also part of emTEX distribution.
29. Reinhard Wonneberger and Frank Mittelbach, 'BIBTEX reconsidered', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 111–124, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
30. Donald E. Knuth, 'The future of TEX and METAFONT', *TUGboat*, **11**(4), 489–489, (1990).
31. Frank Mittelbach, 'E-TEX: Guidelines for future TEX extensions', in *1990 Annual Meeting Proceedings*, ed., Lincoln Durst, pp. 337–345, Providence, (September 1990). TEX Users Group. *TUGboat* 11 (1990/3).
32. Nelson H.F. Beebe, 'Comments on the future of TEX and METAFONT', TUGboat, **11**(4), 490–494, (1990).
33. International Standards Organization, Geneva, *Information processing — Text and Office Systems — Standard Generalized Markup Language (ISO 8879)*, 1986.
34. David Barron, 'Why use SGML?', *Electronic Publishing (EPodd)*, **2**(1), 3–24, (April 1989).
35. Martin Bryan, *SGML: An Author's Guide to the Standard Generalized Markup Language*, Addison-Wesley, Wokingham, England; Reading Massachusetts, second edition, 1988.
36. James H. Coombs, Allen H. Renear, and Steve J. DeRose, 'Markup systems and the future of scholarly text processing', *Communications of the ACM*, **30**(11), 933–947, (November 1987).
37. Reinhard Wonneberger, 'Structured document processing: the LATEX approach', in *Man-Machine Interface in the Scientific Environment. Proceedings of the 8th European Summer School on Computing Techniques in Physics. Skalský Dvůr, Czecholsovakia, 19–28 September 1989*, ed., J. Nadrchal, volume 61 of *Computer Physics Communications*, pp. 177–189. North Holland Publishing Company; Elsevier Science Publishers B.V., (1990).

38. Jaques André, 'LaTeX ou SGML pouvaient-ils faire éviter la catastrophe de la gare de Lyon', *Cahiers GUTenberg*, **1**(1), 21–25, (Avril 1989). English translation in *EP-odd* **2**(3).
39. Lisa M.C. Smith and Mansur H. Samadzadeh, 'An annotated bibliography of literate programming', *ACM SIGPLAN Notices*, **26**(1), 14–20, (January 1991).
40. *TeX Users Group Resource Directory*. Providence, Rhode Island, USA, May 1991. Supplement to *TUGboat* 12 (1991/2).
41. Peter F. Drucker, 'What business can learn from nonprofits', *Harvard Business Review*, **67**(4), 88–93, (1989).
42. Reinhard Wonneberger, LaTeX, Addison-Wesley Kompaktführer, Addison-Wesley, Bonn, 2nd edition, 1988. ISBN 3-89319-152-6 (VVA-Nr. 563-00152-0).
43. Helmut Kopka, LaTeX. *Eine Einführung*, Addison-Wesley, Bonn, 3rd edition, 1991. 340 S., 69,– DM. (Ausführliche Beschreibung von LaTeX ohne SLiTeX).
44. Helmut Kopka, LaTeX*; Erweiterungsmöglichkeiten mit einer Einführung in* METAFONT, Addison-Wesley, Bonn, 2nd edition, 1991.
45. Jost Krieger and Norbert Schwarz, *Introduction to TeX*, Addison-Wesley Europe and Addison-Wesley, Bonn, 1990. Paper, 278 pp, $22.95 translation of [80] by Jost Krieger.
46. Raymond Seroul, *Un petit livre de TeX*, InterEditions, Paris, 1st edition, 1989. ISBN 3-89319-152-6; 317 pp.
47. Rainer Rupprecht, 'Using menudriven TeX under MVS', in *TeX. Applications, Uses, Methods*, ed., Malcolm Clark, Ellis Horwood Series in Computers and their Applications, pp. 81–89, Chichester, (1990). Ellis Horwood. TeX88. Proceedings, Third European TeX Conference. Exeter, England, August 1988.
48. Hope Hamilton, 'Mastering TeX with templates', in *TUG89*, ed., Christina Thiele, volume 10,4 of *TUGboat*, pp. 541–548, (1989).
49. International Business Machines Corporation, *Advanced Function Printing: Data Stream Reference*, 3rd edition, March 1992. S544-3202-02.
50. i-data a-s, *IDA 3820 HP AFP. User's Guide for MVS*, 5th edition, January 1990.
51. International Business Machines Corporation, *Document Composition Facility: Post-Processor Examples*, 2nd edition, January 1991. S544-3484-01, Release 4.0.
52. James Haskell, Wally Deschene, and Alan Stolleis, 'TeX for 30,000', in *TUG89*, ed., Christina Thiele, volume 10,4 of *TUGboat*, pp. 595–602, (1989).
53. Wolfhart Sengler, 'Technische Informationsdienste zur Integration von I-DEAS in die CAE-Umgebung von GM/EDS. Technical information service for the integration of I-DEAS in the CAE environment of GM/EDS', in I-DEAS^{TM}/CAEDS^{R}. *International Conference. Proceedings. 1988 Series*, pp. 325–335, 8 Ill. Structural Dynamics Research Corporation, (Oct. 1988). In English.
54. *TUGboat. The communications of the TeX users group*, **1 ff**(1 ff), (Started 1980). Barbara Beeton (ed.); subtitle for vol. 1–8: *The TeX Users Group Newsletter*.
55. Frank Mittelbach, 'An environment for multicolumn output', *TUGboat*, **10**(3), 407–415, (November 1989).
56. Frank Mittelbach and Rainer Schöpf, 'A new font selection scheme for TeX macro packages; the basic macros', *TUGboat*, **10**(2), 222–238, (July 1989).
57. International Business Machines Corporation, *Document Composition Facility, Document Library Facility: General Information*, 11th edition, January 1991. GH20-9158-10, Release 4.0 for DCF, Release 3.0 for DLF,.
58. International Business Machines Corporation, *About Type: IBM's Guide for Type Users*, 1st edition, August 1986. S544-3122.
59. International Business Machines Corporation, *About Type: IBM's Technical Reference for 240-pel Digitized Type*, 5th edition, November 1991. S544-3516-04.
60. Frank Mittelbach, 'A new implementation of the array- and tabular-environments', *TUGboat*, **9**(3), 298–314, (December 1988). Addenda in [61].
61. Frank Mittelbach, 'A new implementation of the array- and tabular-environments of LaTeX (TUGboat 9#3)—addenda', *TUGboat*, **10**(1), 103–104, (April 1989). Addenda to [60].
62. Christina Thiele, ed. *Conference Proceedings. TeX Users Group. Tenth Annual Meeting. Stanford, August 20-23, 1989*, volume 10,4 of *TUGboat*, 1989.
63. Frank Mittelbach, 'The doc-option', *TUGboat*, **10**(2), 245–273, (July 1989).

64. Donald E. Knuth, 'Literate programming', *The Computer Journal*, **27**(2), 97–111, (May 1984).
65. E. Wayne Sewell, *Weaving a Program: Literate Programming in WEB*, Van Nostrand Reinhold, 1989. ISBN 0-442-31946-0.
66. Adrian Avenarius and Siegfried Oppermann, 'FWEB: A literate programming system for Fortran 8X', *ACM SIGPLAN Notices*, **25**(1), 52–58, (January 1990).
67. Frank Mittelbach, 'Footnotes in a multi-column layout', *TUGboat*, **11**(4), 657–662, (November 1990).
68. Frank Mittelbach and Rainer Schöpf, 'Towards LaTeX 3.0', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 74–79, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
69. Frank Mittelbach and Chris Rowley, 'LaTeX 2.09 ↪ LaTeX 3', *TUGboat*, **13**(1), 96–101, (April 1992).
70. Wolfgang Appelt, *TEX für Fortgeschrittene. Programmiertechniken und Makropakete*, Addison-Wesley, Bonn, 1988. ISBN 3-89319-115-1.
71. Malcolm Clark, ed., 'TEXline. [Nr.1-7:] A newsletter of TEX users in UK and Ireland / [Nr.8ff:] A newsletter of the TEX community.', **1ff**(1ff), (Started 1985). Edited by Malcolm Clark.
72. C.G. [Kees] van der Laan, 'SGML (, TEX and . . .)', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 90–104, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
73. N.A.F.M. Poppelier, 'SGML and TEX in scientific publishing', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 105–109, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
74. Reinhard Wonneberger, 'Approaching SGML from TEX', *TUGboat*, **13**(2), 226–227, (July 1992).
75. Reinhard Wonneberger and Frank Mittelbach, 'SGML. Questions and answers', *TUGboat*, **13**(2), 221–223, (July 1992).
76. Christine Detig, 'TEX & Hypertext — The future of electronic publishing', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 8–12, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
77. L. Carr, S. Rahtz, and W. Hall, 'Experiments in TEX and HyperActivity', in *TEX90 Conference Proceedings*, ed., Mary Gunther, pp. 13–20, Providence, Rhode Island, U.S.A., (March 1991). TEX Users Group. *TUGboat* 12 (March 1991) Number 1.
78. Frank Mittelbach and Rainer Schöpf, 'With LaTeX into the nineties', in *TUG89*, ed., Christina Thiele, volume 10,4 of *TUGboat*, pp. 681–690, (1989).
79. Jan Michael Rynning, 'Proposal to the TUG meeting at Stanford', *TEXline*, (10), 10–12, (May 1990). [Original Title was:] Nordic Proposal to the TUG Meeting at Stanford, August 1989.
80. Norbert Schwarz, *Einführung in TEX*, Addison-Wesley, Bonn, 3rd edition, 1991.