

General Delivery

From the President

Bart Childs

The program of the annual meeting is set. The program committee has done an excellent job. It is appropriate to call this a celebration of 10 years of T_EX and TUG.

I am writing this so soon after the distribution of the most recent TUGboat that I have not had any feedback on the **dingbat** competition. That announcement was missing a statement that a sample submission has been prepared by Doug Henderson and is available from TUG headquarters. I encourage you to submit a dingbat or two. Doug's sample dingbats (an anchor for TUG and a check mark) and methodology should be a big help. Send your entries to Doug; his address is on page 145. Judging will take place at the annual meeting.

See you at the celebration at Stanford.

Editorial Comments

Barbara Beeton

This column seems to be becoming my personal soap box, so I may as well make the most of it. I shall feel free to award compliments where due, and to nag gently when I think it is called for. Mostly, though, I shall try to bring to your attention items that I think are important or interesting, that haven't been mentioned anywhere else in the issue. This will usually include a report on what are the latest versions of the official T_EX-related software, trip reports on meetings I've attended, and anything else that strikes my fancy. Suggestions from you readers are always welcome — the addresses for TUGboat, postal and electronic, are inside the front cover.

T_EX news

The big news about T_EX, really about Stanford, is that the Score machine is to be unplugged on August 31, just after the TUG meeting. This computer has been home to the authoritative T_EX distribution since the beginning. The TUG Board

has been informed of Score's demise, and they are investigating all reasonable possibilities for alternative locations. Several criteria are essential for a suitable new home: the machine must be on the Internet; it must support anonymous FTP; and a T_EX installation must have the full support of the management. All news will be communicated via the electronic mailing lists — T_EXhax, UKT_EX, T_EXMAG, et al. — and published in TUGboat.

As long as Score is viable, updates to the T_EX system will continue to be posted there. The current versions are

T_EX 2.99
 PLAIN.TeX 2.94
 METAFONT 1.7
 PLAIN.MF 1.7

Changes to T_EX, METAFONT and the CM fonts can be obtained by requesting the latest errata list supplement from the TUG office.

All changes have been communicated to all the implementors and distributors on my mailing list. If you are creating a new implementation of T_EX and distributing it to other users, you should be receiving this information. Send me your name and address (preferably an electronic address accessible via the Internet), and a short description of the implementation you're working on.

Updates to items in TUGboat 10 #1, and a suggestion to authors

Several pairs of braces {...} were omitted from Georgia Tobin's article "A handy little font", making it nearly impossible for someone not already versed in METAFONT to recreate the fonts described there. Georgia's column in this issue contains the correct code for the handpointing character.

This brings up the subject of good coding practices. Some readers of TUGboat, I am told, take a new issue directly to their keyboard and enter the code to try it out for themselves. Authors of macros and of METAFONT code can make that activity easier for readers by making their code statements complete and precise. If redundant segments of code are omitted for conciseness, then that should be stated, along with some indication of how the missing code can be reconstructed; the techniques used in Appendix B of *The T_EXbook* may be a good model. Further, an author shouldn't assume that all readers are experts. For example, if @ is used in "internal" control sequences, appropriate `\catcode'\@=...` statements should be included in plain code, or the equivalent `\makeat...` in L^AT_EX.

A couple of years ago, a "Birds of a Feather" session at a TUG meeting discussed what are good

macro coding practices. A number of good ideas were mentioned and recorded, but nothing has been published. I intend to encourage this work to begin again at the upcoming TUG meeting, and perhaps we will see some macro guidelines appear to join the emerging DVI driver standards. In the meantime, the suggestions by Don Hosek for creating portable METAFONT code (page 173) make sense for macros too.

The TUGboat schedule

Unanticipated delays in the printing schedule for issue 10 #1 assured that it arrived too late for most potential authors to respond to the editorial deadline for #2. After considering various possibilities, we have decided to set fixed dates for future issues.

The first issue has always been scheduled to permit authors a break after the holiday rush. In the past, the second and third issue dates have depended on the schedule for the annual meeting; there was sentiment for issue #2 to be in members' hands in time to read before the meeting, and for the deadline for #3 to give sufficient time after the meeting to permit reactions by the authors to what happened there. (The fourth issue—the meeting proceedings—of course depends directly on the meeting schedule, but it has a separate editorial and production staff.)

The new schedule for the three regular issues will follow the same general pattern. For each issue, the editorial deadline will be a Tuesday, permitting authors a weekend for last-minute cleanup. Editorial and production work will be allowed six weeks (the Editor has a full-time job with the Math Society, and TUGboat is her evening and weekend activity). Camera copy will be delivered to the TUG office on a Monday for shipment to the printer. For the past year, the printer has required six weeks; the TUG office is investigating printers and expects to find a reliable one who can handle the job in four or even three weeks.

Editorial deadlines will be determined by the following.

Issue	
#1	3 rd Tuesday of January
#2	2 nd Tuesday of April
#3	Proceedings: as soon as possible, but no later than deadline for #4
#4	2 nd Tuesday after Labor Day

The long summer gap should accommodate a meeting scheduled any time from the middle of July through late August, and also give the Editor some time to relax.

The deadlines for 1990 are given in the calendar; see page 285. Applying the expected production, printing and mailing time, it can be seen that issue #1 may not reach prospective authors until after the deadline for issue #2. To provide an appropriate warning, it has been proposed that the schedule be sent along with acknowledgements for receipt of dues payments. We promise that deadlines will be shown in the calendar for at least two issues.

Informal discussion at the Congrès GUTenberg (see below) yielded some suggestions regarding the technical articles in TUGboat. These included peer review and actual testing of macros and METAFONT code. I believe it is time to consider these actions, as they would enhance the quality and utility of the presentations, but they will mean a change in the way that the proposed schedule is interpreted—the six weeks permitted for production simply doesn't allow the necessary time for communication between reviewer/editor/author should revisions be found desirable. Another suggestion was to include abstracts of the main articles in several languages (English and, say, French and German), also to accept articles in languages other than English. I will solicit opinions and report in the next issue.

So, what are your opinions? Let us hear from you, about the schedule, the suggestions concerning review and testing, or anything else about TUGboat. (If you've noticed no letters to the Editor in recent issues, you've only yourselves to blame.) Electronic mail is, as always, most convenient, and most likely to get a quick acknowledgement, but all comments will be considered carefully.

TUGboat selections on-line

The following additions have been made to the directory <TeX.TUGBOAT> at Score.Stanford.edu since the last issue.

- the tables of contents, file TB1089.CNT, for this year's issues so far.
- FIGPLACE.TeX and FIGSPACE.TeX by Joost Zalmstra and David F. Rogers (TUGboat 10#1)

Check the file -CHRONO-.DIR for a chronological list of the directory contents, and -READ-.TUG and TUGFIL.CHG for a description of the files in the directory and details of changes.

These files are available through August 31 from Score via anonymous FTP on the Internet. Copies have also been installed in the archives at Clarkson and Aston. We are looking into ways to make them available from the TUG office for those who have no network access.

Congrès GUTenberg — A trip report

The second annual Congrès GUTenberg was held in Paris on May 16–17, on the theme “ $\text{T}_{\text{E}}\text{X}$ and graphics”. I was going to prepare a detailed report of the program, but Malcolm Clark has done such a good job (see page 150) that I will limit this report to my impressions.

The first day of the Congrès was occupied by two short courses, the first an introduction to $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, by Olivier Nicole and Jacques André, and the second “first steps in METAFONT”, by Victor Ostromoukhov, which I attended. I had seen some results of Victor’s work in Exeter at $\text{T}_{\text{E}}\text{X}88$, and was pleased to be introduced to some of the basic techniques he uses to make character shapes look their best on a low-resolution display device while enforcing stylistic uniformity within a group of related characters. The letterforms he chose for demonstration were from the outline alphabet known as “blackboard bold”: **R**, **Q**, **N**, **K**, **Z**, **H**, **F** and **C**. (These samples are from the AMS font `msym10`, not from Victor’s production; I feel that he has corrected some of the problems known to exist in these old versions.) These were sufficient to illustrate such principles as uniformity of stem widths and line thickness, keeping paired lines parallel to one another when they are not parallel to an axis, and the like. Victor’s use of a Macintosh to demonstrate, almost instantaneously, both bad and good effects was a powerful reinforcement.

The official first day ended with the annual Assemblée Générale, where the order of the day was the presentation of secretary’s and treasurer’s reports, ratification of actions of the Board, and other items of GUTenberg business. Bernard Gaulle, GUTenberg’s president, presided. I learned that GUTenberg now has about 90 members and another 50 or so subscribers to their journal, the *Cahiers GUTenberg*. If my memory serves, TUG was about that size at the same age; at the age of 10, TUG has over 3,000 members, and I wish GUTenberg the same success. Thanks were extended to several organizations and persons who had contributed facilities or efforts toward strengthening GUTenberg or improving communications among $\text{T}_{\text{E}}\text{X}$ users in the French-speaking world and elsewhere. These included the IRISA and CIRCE (and another, whose name I did not catch) laboratories, the École Normale Supérieure (rue d’Ulm), Peter Abbott (moderator of $\text{UKT}_{\text{E}}\text{X}$), and Pierre MacKay (moderator of $\text{T}_{\text{E}}\text{Xhax}$). Two individuals were made “membres d’honneur”. Raymond Seroul, author of *Le petit Livre de T_EX* was one, and I was very much

honored to be the other. As a token of the honor, I was presented with a copy of *Les bons Romans*, bound from semiweekly editions of 1870–1871 containing the complete *Le Comte de Monte Cristo* by Alexandre Dumas and selections by other authors including de Balzac; when I have finished reading it all, my understanding of French should be very much improved.

The second day, the conference proper, was opened with an introduction by the chairman of the session, Nicolas Brouard. This was followed by greetings from Bernard Gaulle and his recitation of all the anniversaries to be celebrated this year, starting with the bicentennial of the French Revolution. He reported on the concerns of the French $\text{T}_{\text{E}}\text{X}$ community, the various sources of available information, and ended with a review of the business meeting of the previous evening. The presentation of papers then occupied the rest of the day.

Although I am aware that the desire and need for graphics inclusion in $\text{T}_{\text{E}}\text{X}$ documents is very great in many environments, the fact that it is not the most urgent problem facing the Math Society has insulated me from most such activity. The scope and ingenuity of the approaches described by the speakers was most impressive. Many tools are available to assist in document preparation, and of these, quite a few are compatible with $\text{T}_{\text{E}}\text{X}$, though the graphics tools are usually dependent on the facilities of particular output devices. Nelson Beebe observed that the basic problem is that $\text{T}_{\text{E}}\text{X}$ came ten years too soon. (Of course, if it hadn’t, we wouldn’t be able to celebrate ten years of $\text{T}_{\text{E}}\text{X}$ this summer, and the Congrès GUTenberg wouldn’t have taken place.) PostScript was mentioned in a number of contexts, with or without connection to the Macintosh. Another topic that kept surfacing was that of standards, graphics and otherwise. Standards seem to have generally stronger support in Europe than in the U.S., although with the adoption of SGML by the Department of Defense the awareness of standards has been raised considerably. The interoperability of $\text{T}_{\text{E}}\text{X}$ and various document and graphics standards seems a very fruitful area for investigation. Several papers described more individualistic solutions to particular problems; the use of the “screen graphics” symbols to produce diagrams was one such. Another suggested approach was to reprocess DVI files to insert commands suitable for particular print engines. DVI portability is sometimes hard to realize even with text, and the problems are magnified with graphics. One goal that seems worth working for is the ability to archive documents for long-term storage; if graphics are to

be an integral part of such archived documents, then we might be warned to think hard before revising \TeX too soon.

I have already mentioned Raymond Seroul and his book, *Le petit Livre du \TeX* . I would like to recommend this book highly (and not just because Raymond gave me a copy). It contains, among other useful features, a fine "Dictionnaire-Index", what I would call a glossary, which lists, with extensive explanations and references to the main text, all the important control sequences and other useful concepts, e.g. "acolade" (a brace), "fonctions (noms de...)". I have heard this kind of reference suggested many times, but *Le petit Livre* has the first example I have seen in print. TUG is investigating having it translated and making it available in English. In the meantime, the French edition should present little difficulty, and much information, to a reader who has some familiarity with the French language, and a dictionary.

Raymond Seroul,
Le petit Livre de \TeX ,
 Préface de Dominique Foata.
 InterEditions, Paris, 1989. 317 pp.
 ISBN 2-7296-0233-X

Finally, I can also recommend the *Cahiers GUTenberg*, which is a window on the \TeX world with quite a different view than TUGboat. Malcolm Clark's report on the Congrès gives all the details of how to join GUTenberg and subscribe to the *Cahiers*.

Réflexions sur le Congrès GUTenberg Paris, Mai 16-17, 1989

Malcolm Clark

The French \TeX users group has been around for a few years, in an 'unofficial' form. Over the last year or so they have become 'official', much more active and they now organise an annual meeting. The first well-publicised GUTenberg meeting was held in Paris last year. I was impressed there by the attendance (well over 100), the stamina (the room was tiny and without air-conditioning), and the range of topics covered. There is, in any case, a tradition of \TeX activity in France — the second European \TeX Conference was held in Strasbourg, in 1986.

This year's meeting was again in Paris, but used far larger rooms to accommodate the 120-150 people who attended. Scanning down the list of attendees, there are the usual academic and research organisations, but also publishers, and the printing trade in general. This bodes well for the future. The meeting was held over two days: the first day was given over to two seminars — one on \LaTeX (from Olivier Nicole and Jacques André), and the other on METAFONT (Victor Ostromoukhov); followed by the AGM. The second day was the conference proper. I attended part of Victor's 'Premiers pas en METAFONT'. As usual, I was impressed by Victor's breadth and depth of METAFONT-lore. And he seemed to be getting something useful across to the forty or so would be METAFONTers. Since he used a Macintosh to demonstrate the points, there was a reasonably quick interaction between intention and realisation. I confess I didn't stay to all of this; my powers of concentration are not great enough to follow a full day of technical METAFONT (far less in a foreign tongue). However, one quote from Victor:

```
<mathematical typesetting> → <empty>|< $\TeX$ >
< $\TeX$ >→< $\TeX$ 82><font support>
```

That sums things up nicely I think.

Bernard Gaille, GUTenberg's President ran the AGM with great efficiency and some humour. I particularly like the French style of democracy (it runs: question — 'anyone against?'; answer — 'no'; conclusion — 'passed'; excellent). I won't plough through all the bits of the AGM, except to note that the group is in excellent financial health, that this is a year of anniversaries in France — 200 years since the Revolution, but also some others: 10 years of TUG, 50 years of CNRS (Centre Nationale de la Recherche Scientifique), one of the homes of \TeX , and of course this is also the year of the 4th European \TeX Conference. The AGM honoured two people with 'honorary membership': Barbara Beeton and Raymond Seroul (the author of *Le petit Livre de \TeX*). By way of recognition, they were each presented with an edition of 'Les bons Romains', published over a hundred years ago. Peter Abbott was also thanked for the help he has given in easing '[les] perturbations EARN/Bitnet'. I was particularly pleased to see Barbara honoured and Peter thanked in this way. The whole \TeX community owes them much for their dedicated adherence to the cause, and it was particularly refreshing and tactful that GUTenberg saw fit to include them in this way.

Besides this conference, GUTenberg produces its own journal, *Cahiers GUTenberg*. The inaugural (or prototype) edition (confusingly numbered 'zéro') was available at last year's conference. The first and

second editions were out by this year's conference. Many of the talks in the conference were also printed in the *Cahiers* (which helped me enormously). Although the group is 'francophone', several articles are in English. Allowing for the technical words which dictionaries never seem to get right, it isn't too difficult to make sense out of the papers/articles. The *Cahiers* represent a major undertaking, in time, effort, and in financial commitment (as I well know from my own limited venture in *TeXLine*). If the high standards already being established are maintained, GUTenberg will have created something which will be of great and lasting service to the whole *TeX* community. My only minor criticism of the *Cahiers* is the lack of consistency in the provision of abstracts (a failing it shares with TUGboat). If abstracts were included, it would be possible to prepare multilingual translations which could be circulated more widely, alerting others to the range and relevance of the material.

The major theme of the Conference was 'graphics' — a popular one these days. Fortunately, thanks to Sebastian Rahtz' talks, I think I know a little about the background here. Rather than report each talk in detail, it is perhaps more productive to try to select some of the major themes. After all, the text of most of the talks is available. As usual, it is notable how far *L^AT_EX* dominates in Europe (or perhaps, just outside the US). It is also notable that 'standards', however defined, keep cropping up: X-Windows, POSTSCRIPT, *TeX* itself, PHIGS, GKS, SGML, and even emacs. This seems particularly healthy, although equally there are many forays into areas which are less portable. Nevertheless, the apparent domination of C, as the implementation language of choice, (with or without the spectre of Unix) would indicate the possibility of transferring some of the applications to other platforms. From the summaries, it is evident that one of the great concerns is the use of POSTSCRIPT, and the incorporation of POSTSCRIPT (and EPSF) files into (especially) *L^AT_EX*. Perhaps my favourite paper was Maurice Laugier's. His was a very simple and straightforward idea — namely that the PC's graphics characters may be mapped quite easily into rules, and that tables (and some diagrams) may be prepared by this means, provided that a monospaced font is adequate.

A list of the talks, together with a brief summary (usually the authors'/author's own) is included here. Some of the talks were in English (a bold move for a francophone group), and some summaries (notably that of Lance Carnes) were distributed in English and French. Very tactful.

L'année de tous les anniversaires: Bernard Gaulle. A welcome to GUTenberg, and an overview of the services and facilities available to *TeX* users in France (and elsewhere); touches on the public, private and commercial domains; addresses GUTenberg's relationship with the rest of the world.

Xwindows, L^AT_EX, T_EXdraw et Plot79, ou comment calculer, rédiger, dessiner et imprimer plus aisément: Nicolas Brouard. A workstation running under X-Windows offers a way of calculating, writing and designing which is much simpler than with a 'classic' terminal. Two graphics tools, *TeXDraw* (a public analogue of MacDraw) and *PLOT79* (a 3D graphics system based on CORE), can easily be employed with *L^AT_EX*.

TeX and Graphics: the state of the problem: Nelson Beebe. Inclusion of graphics in documents typeset by *TeX* is not yet a satisfactorily solved problem, and no final general solution is in sight. This paper surveys alternatives for insertion of graphics in *TeX* documents. It summarizes graphics primitives of several modern software systems, and shows how *TeX* has seriously deficient support for their direct incorporation in *TeX* itself.

L'environnement de production de documents T_EX à l'IRISA: Philippe Louarn & Bertrand Decouty. The group consists of over 200 researchers, teachers and engineers, who need to produce reports, articles, theses, books... The objective was to provide a set of homogenous tools in a heterogenous environment (Unix, VMS, workstations, PCs...). Naturally, *TeX* was chosen as the fundamental tool. In graphics, the two main avenues are through the incorporation of POSTSCRIPT, and by the use of eepic in the *L^AT_EX* picture environment.

Survey of T_EX Graphics for the PC: Lance Carnes. This talk reviews the various graphics systems offerings for the IBM PC and compatibles, and the ways these systems can be used in conjunction with *TeX*; the use of POSTSCRIPT, HP PCL and bitmap files in conjunction with current *TeX* drivers; the use of output from screen oriented drawing systems; and the conversion of graphics files from one format to another, and scaling of images, for inclusion in documents.

TeX et les graphiques dans l'environnement Mac: Anestis Antoniadis. Painting and drawing are the two sides of the creation of graphics on the Macintosh. *Paint* images (otherwise known as bitmaps) are known to the Mac as a set of points on the screen. *Draw* images (also known as

vector drawings) are known to the Mac as objects (rectangles, lines, circles, polygons) and are defined by their mathematical attributes. As a consequence of the way in which they are defined, they take full advantage of the resolution of POSTSCRIPT peripheral devices. The goal of this article is to give an overview of the methods and software for the generation of graphics on the Mac, and to discuss the insertion of such graphics in documents prepared with *Textures*, one of the implementations of T_EX on the Macintosh.

METAFONT et POSTSCRIPT: Victor Ostromoukhov. Conversion between METAFONT and POSTSCRIPT is possible. What are the best techniques, and what are the constraints? (No written contribution, but see his MacMETAFONT program.)

DDI: un environnement de travail pour la réalisation de graphiques scientifiques, techniques et fantaisies utilisables avec T_EX: André Violante. DDI is a work environment for the creation of scientific, technical and artistic graphics. The fundamental idea of the system is the creation and use of graphic fonts. To use these, several tools are available: design software (*Designcad*); a program to convert *Designcad* files to METAFONT; METAFONT itself; G_FtoP_K; T_EX; and a suitable device driver.

texpic: design and implementation of a picture graphics language in T_EX à la pic: Rolf Olejniczak. **texpic** is a T_EX implementation of a graphics language similar to Kerhighan's *troff* preprocessor *pic*. The implementation consists of two parts, a set of elaborate T_EX macros and a postprocessor for drawing (in the *dvi* file). **texpic** objects and T_EX or L^AT_EX commands may be combined at will. **texpic** is written in C, and is fully portable, to the extent that every T_EX implementation, every preview and every correctly written printer driver will work with **texpic**.

Traduction en T_EX d'un fichier SGML avec récupération des graphiques et des tableaux: Maurice Laugier. The logic of SGML markup is close to that of L^AT_EX, and translation from SGML to L^AT_EX can be realised quite simply. However, the problem is rather different for graphics and tables. This paper presents a way in which the PC graphics characters were successfully mapped into L^AT_EX, easing the production of tables and simple diagrams.

GIT_EX, PAPS: deux logiciels manipulant POSTSCRIPT et L^AT_EX: Christophe Cérin. From PC-based graphics programs, PAPS (Programme

d'Application POSTSCRIPT) transforms an image into a POSTSCRIPT format. It will also allow some manipulation of the graphics image. GIT_EX (Générateur d'Image T_EX) is a program which allows a *figure* environment to be constructed for inclusion of the POSTSCRIPT into a L^AT_EX file.

L'incorporation de graphiques dans INRST_EX:

Michael Ferguson. The approach used in INRST_EX is to use the power of the printer to create graphics, and not to introduce special characters. The capability to generate graphics has been achieved for POSTSCRIPT and for QMS graphics on a QUIC laser printer. The paper discusses the role of the printer as well as the need for support software to permit the incorporation of graphics produced by other systems. The paper also discusses some of the limitations inherent in the choice of graphics systems.

Like most conferences, the most interesting and valuable discussions take place in the corridors, over coffee, or at lunch. This tradition was maintained here. The corridors were also used to display various pieces of T_EXware. The inclusion of a noticeboard for general T_EX-notice, trivial as it seems, was extremely useful. There was a display of ArborText's *Publisher* (about the only place you don't see *Publisher* these days is in the UK), and an extensive display of books, where Raymond Seroul's book was selling well (on its first day of publication).

Conclusion

I was impressed. There can be no doubt about it, GUTenberg provides us all with much to emulate. The strength and coherence of the group is manifest. Taken purely at the national (or francophone) level, GUTenberg is contributing massively to the strength of T_EX. The *Cahiers* are excellent, and look capable of sustained quality. The annual meeting is now well-established and imaginative. GUTenberg's involvement with several French publishers must also be a good sign.

GUTenberg is also keenly aware of other French speaking areas (Belgium, Switzerland, Quebec) and of the advantages of international electronic communication. And the committee ensured that the various national representatives (myself, representing TUG and UKTUG; Joachim Lammarsch, representing the German group, Dante; and Kees van der Laan representing the Dutch group) had the opportunity to discuss how we could cooperate for the common good.

I am a shade worried by what I see as the determination to stay outside the TUG orbit. Perhaps I am over-sensitive — being described either as English or Anglo-Saxon does tend to make me a trifle testy — but I have always thought of TUG as an international organisation, not an American one. Others do not share this perception. If our conclusion is that we are not getting what we need out of TUG, the solution is in our own hands. We can influence the organisations in which we participate. That's the key — participation. T_EX must be worth it.

Joining GUTenberg; subscribing to the Cahiers

To join GUTenberg, you need only part with 200 FF. This has two advantages (besides preparing you for 1992 and demonstrating your adherence to the European ideal) — it enables you to pay a reduced fee at the annual meeting and for the *Cahiers*. To obtain the *Cahiers* costs a further 150 FF if you are a member, but 250 FF if you are not. This year's conference cost 200 FF for members and 400 FF for non-members. Clearly membership pays for itself if you are contemplating attending the conference and taking the journal. In other words, joining GUTenberg and subscribing to the *Cahiers* costs you a total of 350 FF (made payable to GUTenberg). Of course Eurocheques are acceptable.

Note that membership comes in several different categories: individual membership is 200 FF; institutional membership on behalf of a non-profit organisation is 700 FF; while institutional membership on behalf of profit-making (as opposed to *profitable?*) organisation is a hefty 1400 FF. On the other hand, institutional membership does allow you to nominate up to seven individuals.

Send your money to:

GUTenberg

% IRISA

Campus Universitaire de Beaulieu

35042 Rennes Cedex

France

◊ Malcolm Clark
Imperial College Computer Centre
Exhibition Road
London SW7 2BP, England
Janet: `texline@uk.ac.ic.cc.vaxa`

International Standards and T_EX

Malcolm W Clark

Editor's note: The following "article" is really two papers that Malcolm prepared as notes for his presentation on standards at the June Nordic T_EX meeting (see page 287). International standards are becoming increasingly important in technical publishing, and they will undoubtedly affect the way in which many T_EX users carry on their work. It seems only fair that we should work to make T_EX affect the way in which standards are defined. With Malcolm's permission, these papers have also been submitted to the U.S. working groups on the Office Document Architecture (ODA) standard, X3V1.3 and .5.

Standards and T_EX

T_EX is a standard. It is rigidly defined; every implementation of T_EX must pass the 'trip' test before it has the right to call itself 'T_EX'. Even the components of T_EX are standards; dvi format is defined rigorously; even the formats of p_x1, p_k, and g_f are defined. Thanks to this standardisation, every user of T_EX (L^AT_EX and $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX) knows that he or she can expect the same results from the same input, no matter what equipment they use to prepare their document. This is a degree of standardisation which is unheard of outside the T_EX world.

There are areas within the T_EX world without adequate standards. Some of these have been at least considered by the T_EX community. Both device driver standards and macro-writing standards have been the subject of working parties: the driver standards working party has made some preliminary announcements, but the macro-writing standards working party seems to be moribund. Other areas require consideration, including the handling of *specials*, but this may be tackled by the driver standards people.

But there are wider standards which affect us in the T_EX world. The T_EX world is only a small (and many would argue, privileged) part of the 'document' world. There are international and *de facto* standards which are of critical importance to us.

We have to acknowledge the importance of the *de facto* standard, POSTSCRIPT. In a sense, T_EX 'sits above' POSTSCRIPT. No-one (well, almost no-one), would ever dream of writing a document in straight POSTSCRIPT. Normally, we write a program which generates POSTSCRIPT — for example,

TEX. Nevertheless, we must be aware of the way in which POSTSCRIPT-compatibility is crucial if we are to be taken seriously by the rest of the world. We must have an acceptable answer to the question 'Can you generate POSTSCRIPT?', even if we feel that the question is ill-posed.

The first 'international' standard which is of importance to us is SGML, the Standard Generalized Markup Language. TEX is itself a markup language, but SGML takes this one step further to become divorced completely from the realm of typesetting. L^AT_EX has a closer affinity to SGML, although it does not go quite far enough. While it is possible to argue that the basic paradigm of SGML is flawed, its widespread acceptance and use (by, among others the US DoD, and the EEC) demands that we do not ignore it. Many SGML-based systems use TEX (or L^AT_EX) as the document formatting engine.

But SGML is an existing ISO standard. An evolving standard of which we must be keenly aware is ODA (Office Document Architecture). One of the objectives of ODA is to permit the electronic interchange of documents over open systems. In the TEX world we would argue that this has already been achieved. One aspect of concern to ODA is mathematical text. Again, we would argue that this has already been achieved. Sadly, the way that national and international standards are created does not ensure that the best *de facto* standard becomes enshrined in the ultimate ISO standard. At present, the various ODA national committees and panels are considering the input of mathematics. The principal European submission which has been received suggests the use of eqn.* This is a somewhat limited and limiting approach. It is of the utmost importance that we in the TEX world promote the other alternatives that we know (and love).

We can live outside the 'standards world'. It is possible. But it is uncomfortable, and ultimately it will lead to atrophy. We should ensure that decisions, like those to become part of the ODA standard, are made with reference to a wide spectrum of possibilities. It may be that TEX is inappropriate to ODA. But that conclusion must be reached by active and informed debate, not by ignorance and apathy.

* Editor's note: A U.S. contribution has recommended examination of eqn, TEX, and SGML.

An approach to the interchange of mathematical expressions

Introduction:

There are a number of existing ways in which mathematical information may be interchanged, using only the ASCII character set. These include eqn, TEX (and its siblings, L^AT_EX and A_MS-TEX), and SGML. There are, of course others, but they tend to be linked to some proprietary system. The three above all have the advantage of belonging, in a sense, to no-one. The first two, eqn and TEX have the further advantage of having been tested 'in the field' for a number of years. They are well understood by a large population. eqn is, of course, a Unix tool, distributed widely with that quasi-standard operating system. TEX has been implemented on a wide variety of computers (including Unix machines), and has the further advantage of belonging even more firmly to the public domain—to the extent that its algorithms often crop up in proprietary systems. Mathematical encoding in SGML is possible, but is found infrequently, many SGML systems opting out of the difficulties by adopting either eqn or TEX as their mathematical processor.

It is not the purpose of this presentation to extol the virtues of TEX over eqn. The purpose is to make evident the power of TEX to encode mathematical expressions in an unambiguous and straightforward way which is generally both 'human-readable' and coherent, and which may also be reasonably compact. And by implication, that any consideration of a suitable technique for the interchange of mathematical information should include examination of TEX.

It should be understood that 'TEX', as used here, is intended to include the common 'add-on' facilities provided by both L^AT_EX and A_MS-TEX.

Use of existing standards:

As noted above, TEX uses only the ASCII character set; thus it is commonly used for the transmission of technical material over existing local- and wide-area networks. This practice is some years old now. TEX, (unlike eqn), has no 'reserved words'. The sole 'reserved' character is the backslash \ (and even that is not very reserved), which is used as an 'escape character' to denote that the token (symbols) which follow should be treated in a special way. Thus \alpha is a way of representing α . This takes us no further than SGML, where we could use the Public Greek Symbols Entity 'alpha'. However, it does provide us with extra tools which enable us

to write expressions like $a \over b$ to obtain $\frac{a}{b}$. It is but a small step from this to something a little more grandiose like

$$A = \prod_{j=1}^{p-1} \int_0^\theta L_j = \theta^{p-2}$$

or

$$\frac{\phi(\theta_i) - \phi(\theta_{i-1})}{1 - \phi(\theta_{i-1})} = P, (i = 1, 2, \dots, n)$$

Keyboard symbols are used when appropriate (like the parentheses, the = and - signs), but all other symbols are obtained through the use of the \ operator. The T_EX encoding for these two expressions is:

```
A = \prod^{p-1}_{j=1}\int_0^\theta
  \d L_j=\theta^{p-2}
```

and

```
{\phi(\theta_i) - \phi(\theta_{i-1})
  \over 1-\phi(\theta_{i-1})}
 = P, (i=1,2,\ldots,n)
```

Apart from explaining that the ^ and _ symbols are used to denote super- and sub- scripts, the only other point to note is the use of { and } for grouping sub-expressions.

Parenthetically, an apparent ambiguity:

Obtaining superscripts requires comment. T_EX uses the same ‘operator’ for superscripting and for ‘raising to a power’: that is to say, x^2 , where the ‘2’ is a power is indistinguishable from x^2 , where the ‘2’ is a ‘true’ superscript (or superior) — they are both obtained from $x^{\wedge}2$. This may be seen as an ambiguity. If it is, we can point to the T_EX control sequence \sp which can be used in place of ^ . Thus, although $x \sp 2$ will give exactly the same formatted result as $x^{\wedge}2$, they would be distinguishable in the original ASCII text. In common with most computer systems, there is always a tendency to want to minimise keystrokes, so that most T_EX users would tend to type ^ rather than \sp, even if what they meant was ‘superscript’.

Extensibility:

Indirectly this also points to another feature of T_EX. It allows the creation of ‘macros’ — combinations of more primitive commands which can be extended to provide very powerful features. As an example, we may take \matrix, a macro which allows us easily to write expressions like:

$$A = \begin{pmatrix} x - \lambda & 1 & 0 \\ 0 & x - \lambda & 1 \\ 0 & 0 & x - \lambda \end{pmatrix}$$

where the T_EX code is

```
A=\left(\matrix{x-\lambda & 1 & 0 \cr
  0 & x-\lambda & 1 \cr
  0 & 0 & x-\lambda \cr
}\right)
```

For a fuller explanation here we need to add that & is the ‘pseudo’ tab character, which separates columns, and \cr is a control sequence which indicates the end of a row.

T_EX is, in fact, a powerful programming language its own right, although this may not be an essential feature in this context. Nevertheless, it indicates that any apparently omitted features in the base language may be created for future needs, or, in fact, needs outside strict mathematical uses.

A de facto standard:

T_EX has been used widely in the academic community (the main users of mathematical typesetting software) since 1978. It was first released in a prototype form in 1978, but underwent a substantial revision before its present form, which was completed in 1982. Since then, T_EX has been ported to an astonishing variety of computers. Almost any and every machine with at least 16-bit words has a version of T_EX working on it. All versions of T_EX must pass the so-called ‘trip’ test, before they may call themselves T_EX. This is intended to ensure that each and every version produces exactly the same output for the same input.

Naturally it is not the formatting qualities of T_EX which are relevant here, but instead its ability to represent the full range of mathematical expressions. One indication of its success in this area is the use to which T_EX has been put in teaching blind mathematicians. We know of at least two projects where this has been undertaken, with apparent success. The attractiveness of T_EX here lies in its ability to linearise an expression, much as one would do if one had to ‘talk’ maths to a colleague, without the advantage of chalk and board.

But besides this linear quality, we must point to the widespread use and adoption of T_EX, the de facto standardisation which has occurred in the academic world, and the huge pool of T_EX-familiar keyboarders.

Conclusion:

T_EX has the ability to encode mathematical expressions in a way which is both human-understandable and easily transmitted by electronic means. A great many implementations exist which can turn such encoded material into displays of one sort or another — the linear encoding is readily transformed

into a two dimensional display. Since \TeX has been available widely for a number of years, it is well understood by a large population, although, as public-domain software it lacks the outright backing of any large commercial organisation.

The approach adopted by \TeX , and the lessons which may be learned from it, can be put to good use as part of the ODA standard for mathematical encoding. It is not the contention here that only \TeX is a suitable vehicle, or even that adoption of \TeX as it exists now is the very best solution. The principal argument is that to exclude it from discussion and examination would be counter-productive and limiting.

◊ Malcolm W Clark
Imperial College Computer Centre
Exhibition Road
London SW7 2BP, England
Janet: `texline@uk.ac.ic.cc.vaxa`

Teaching \TeX

Bart Childs

Acknowledgements, Disclaimer, and Calls

This is a continuation of the paper on "Syllabi for \TeX and METAFONT courses", pages 117-127 in the Proceedings of the 9th TUG meeting. This paper contains some of the details of the previous paper in tabular form. The errors are mine, but many of the ideas are those of the teams who did the work. I would especially like to thank Barbara Beeton, Dean Guenther, Pierre MacKay, and David Ness for their continual reading, editing, and other input. Don Knuth also reviewed an earlier draft and said he is no longer a "10". Well ...

This paper includes a flow chart to illustrate the prerequisites of \TeX , \LaTeX , and METAFONT classes. A two page table is used to show the contents of the \TeX courses, and another table shows the contents of the \LaTeX courses. The \LaTeX overview was not written by \LaTeX experts. The three level test for \TeX has been revised several times.

I am **calling** for input of the following form:

1. constructive criticism of this document, especially the \LaTeX contents;
2. submission of additional questions for the \TeX tests;

3. submission of questions for the (yet to be done) \LaTeX tests; and
4. your suggestions for any other items that you think will help.

I will make these sources available for all. I intend to incorporate input and reissue this document on a periodic basis. The rest of this paper is what I would distribute.

Overview

A user should be familiar with the use of a text editor before undertaking the *Beginning \TeX* course. In some cases the user can employ a word processor and store the file as an ordinary text file if such an editor is the user's common means of creating a file.

The *Beginning \TeX* course should give the student an understanding of the basic nature of \TeX and the parameters it uses in producing attractive documents. After the course, the student will feel comfortable taking examples from *The \TeX book* for use, but may not yet be fully at ease modifying these examples.

Upon completion of the *Intermediate \TeX* course, the student should be able to adapt and modify examples from *The \TeX book* to suit individual purposes. He or she will also be able to develop creative solutions to typesetting problems using \TeX .

The *Advanced \TeX* course should give the student knowledge of how many of the examples in *The \TeX book* are created. Further, the student should be able to create new macros and documents using these concepts. At this stage of knowledge, \TeX 's capability as a 'text-oriented programming language' can be exploited.

Beginning \TeX

This course provides a practical introduction for those with limited, or no, exposure to \TeX and will be composed of about equal parts lecture and "hands-on" sessions, including many practical exercises for each object of study. Participants will be introduced to \TeX as a language for typesetting, also learning its context in the history and milieu of word-processing and typesetting. \TeX is compared with other popular formatting systems such as word-processors and desktop publishing systems.

\TeX concepts to be covered include: methods of preparing simple paragraphs, changing line spacing and specifying fonts; simple boxes, characters and accents; justification and line breaking. In math mode, superscripts, subscripts, and fractions will be addressed.