General Delivery

From the President

Nelson H.F. Beebe

I have just received *TUGboat* 10, no. 4, which is the Proceedings of TUG's Tenth Anniversary Conference; what a fine issue! And 320 pages long, too. This is the first Proceedings that has appeared as a regular issue of *TUGboat*, a practice that we hope to continue. The TUG community is an exciting one to be a part of.

It is really satisfying to see such a broad range of topics covered, from fractals to Thai typesetting, from Icelandic dictionaries to the Encyclopedia of Mexico. I'm sure Don Knuth is amused that TEX is even being used to write computer games.

Malcolm Clark's survey of "Olde Worlde" (European) TEX activities is a welcome one. The French-speaking group, GUTenberg, has published several issues of *Cahiers GUTenberg*, and will host its third annual conference in Toulouse in May, 1990. The German-speaking group, DANTE, has just released the first issue of its newsletter, *Die TEXnische Komödie*.

I'm happy to report that, under the capable editorship of Dr. A. van Roggen, the journal *IEEE Transactions on Electrical Insulation* is now set entirely with LATEX. The 330-page issue that I have just received is set in two-column format with, it seems, about one figure per column, and sometimes several. The formatting requirements often raise difficult questions, and the editor has kindly provided me with a long list of comments that I hope can be addressed in the LATEX developments discussed below.

In early January, I undertook a substantial rearrangement of the TEX directories on our local facility, which includes TOPS-20, VAX VMS, UNIX (several flavors), IBM PC DOS, and Apple Macintosh systems. My motivation for this was to remove variations in directory structure, so that our users can easily find files in the TEX tree. With the help of TOPS-20 FTP and several UNIX utilities, it has been possible to preserve exact time stamps across most of these systems, greatly easing the job of determining when files are out-of-date with respect to the master archives.

The largest impact this reorganization had was on UNIX TEX, and on the font directories. UNIX TEX has traditionally dumped TEX, LATEX, AMS-TEX, and BIBTEX input files all into a single directory, despite the fact that the implementation has always supported environment variable search paths. After the reorganization, there are now separate subdirectories for each of these systems. Fonts had been stored in a single directory too, which was probably all right when we had only Computer Modern fonts. After the reorganization, I now have separate directories for AMS, Computer Modern, Concrete, Euler, Greek, Imagen, music, Pandora, and AP-TEX font families.

It seems to me that it would be worthwhile to try to develop a recommended TEX file tree organization that would promote standardization in the entire community. To that end, I invite site coordinators in particular to send their comments to me, and to supply me with a summary of the TEX file tree layouts that they are currently distributing.

Kinch Computer Company announced at the Stanford TUG meeting the availability of PostScript fonts in 300-dpi pk file format, which are tentatively being called AP-TEX fonts. We licensed a copy of these, and I have developed a set of style files for TEX and LATEX that make it trivial for our users to switch from Computer Modern to some other font family by means of a single TEX `\input` statement or a LATEX `\documentstyle` option. After a suitable test period, I expect that these style files can become generally available.

Note that having PostScript fonts in pk file format means that they can be used with almost any DVI driver and any output device (assuming that fonts of the required resolution are available), including screen previewers like `xdvi` in the X Window system on workstations. This solves an important problem of how to proof phototypesetter output on low-resolution devices when fonts other than Computer Modern are required.

The 29 January 1990 issue of *InfoWorld* on pages 46–47 carried an advertisement for a series of controller boards for IBM PC and Macintosh II systems that can turn 300-dpi Hewlett-Packard LaserJet Series II and Apple LaserWriter II printers into 400-dpi, 600-dpi, 800-dpi, and 1000-dpi PostScript printers that the vendor terms "plain paper typesetters". If these are found to work satisfactorily, it seems to me that this announcement represents a very important advance for the TEX community. I would very much like to hear from anyone who tries any of these products. I am curious about the extent to which toner granularity and paper roughness impact the apparent output resolution.

In *TUGboat* 10, no. 3, I outlined some projects that I'd like to see accomplished during my time in

TUG office. The first of these was the improvement of electronic retrieval of TEXware, particularly for the many users who have electronic mail access, but no Internet FTP access. As the new books of Frey and Adams [2] and of Quarterman [3] show, electronic mail can now reach several countries in Asia, and even a few in Africa. I hope that the recent political changes in Eastern Europe open paths there as well.

I have implemented a modified version of the NETlib system (see [1]). It is tentatively called TUGlib, and is operational on a local machine, providing access to the TEX collections on `science.utah.edu`; it is not yet available to the public. TUGlib could be moved to the home of the TEX master archives. Michael DeCorte at Clarkson University has independently implemented an excellent e-mail server for the Clarkson TEXware collection. Each of these systems has features the other lacks, and further investigation is needed before we make a choice that will be offered to the public.

I expect to soon have the TUG address list for extraction of electronic mail addresses to facilitate constructing an automatic forwarding mechanism for TUG members, permitting mail to be sent through a standard site when you do not know the exact electronic mail address of a TUG member.

The second issue was that of trademarks, which impacts us now that Intel Corporation has claimed a trademark on the name **DVI**, which they think stands for "Digital Video Interface", but of course, TEX users know what it really means. We have now initiated legal negotiations with Intel about this.

The third issue was the relationship of national or language user groups to TUG. Malcolm Clark's survey cited above should be helpful in understanding the background here.

The sixth issue was the future of LATEX. The outstanding work of Frank Mittelbach and Rainer Schöpf reported in the Proceedings has been followed by their establishment of an electronic mailing list intended to reach a small number of interested people. This forum should provide a means where the critical issues in LATEX development can be discussed by geographically distributed experts and wizards.

I expect that 1990 will be an interesting year for TEX users, with these conferences scheduled so far: a DANTE meeting in March; the GUTenberg meeting in May; the annual TUG meeting in June at College Station, TEXas; an NTG-SGML meeting in August; TUG's first meeting outside North America in Cork in September; and another DANTE meeting in October. See the calendar for details.

References

[1] Jack J. Dongarra and Eric Grosse. Distribution of mathematical software via electronic mail. *Comm. ACM*, 30(5):403–407, May 1987.

[2] Donnalyn Frey and Rick Adams. `!%@`*: A Directory of Electronic Mail Addressing and Networks*. O'Reilly & Associates, Inc., 981 Chestnut Street, Newton, MA 02164, 1989. ISBN 0-937175-39-0.

[3] John S. Quarterman. *The Matrix*. Digital Press, 1989. ISBN 1-55558-033-5.

◇ Nelson H.F. Beebe
Center for Scientific Computing and
Department of Mathematics
South Physics Building
University of Utah
Salt Lake City, UT 84112
USA
Tel: (801) 581-5254
Internet: `Beebe@science.utah.edu`

From the Past President and Annual Meeting Host

Bart Childs

We have finally received numbers 3 and 4 of volume 10. They are beautiful and well done. Please don't be critical of our staff and volunteers because it is often our procrastination and the dealing with the real world that cause the delays.

Barbara has done her usual excellent job with the regular issues and Christina's job in getting the Proceedings of the Annual Conference out as number 4 was frustrating to her but a great contribution to us. We can never say thanks enough. [Editor's note: Barbara wishes to acknowledge the skillful contribution of Ron Whitney to this whole enterprise. Without his assistance, everything would take much longer and be much less well organized. Thanks, Ron!]

I was particularly pleased with Nelson Beebe's article. He addressed several areas that we should consider for the future of TUG. I think this shows the value of change in such societies. I encourage each of you to be more involved in TUG; your membership and active participation helps each and every one.

We Aggies are looking forward to hosting the Annual Meeting here in June. Ray has scheduled a Board of Directors meeting for Sunday. This should allow us to get most (hopefully all) of our businesss done and then use the lunch time for interaction with the membership. We have been frustrated and rightly criticized by not being available to meet with the rest of the membership.

Sunday evening we are going to have a TEXas ethnic feast. The highlight will be TEXas BBQ, TEXMEX, something to drink, Czech pastries for dessert, and Vrazel's Polka Band. The first two items of food are not 3-alarm type and should not cause anybody a problem. We plan to have some *jalapeños* available for anybody who wants to show (make your appropriate selection) their:

□ fondness for hot peppers,
□ virility,
□ lack of taste buds, or
□ lack of judgment.

The musical group plays music from *Urban Cowboy* type to ballroom. Most of it is in between and comes from the rich local heritage that is based upon immigration from western Europe, particularly Czech and German. We will have several couples of designated dancers who will dance with willing brave souls. They do a large number of specialty dances with titles like: the *Herr Schmidt* which is also known as the *Mexican Hat Dance*, the *Chicken Dance* which we will do especially for Ray Goucher, *Schottische*, *Cotton Eyed Joe*, ... It will be a great time for visiting as well because they use their amplifiers with discretion.

I hope to participate in at least three birds-of-a-feather sessions. The topics that I think need addressing include:

1. What should be in a TEX distribution, especially which tools, fonts, magnifications, METAFONT, ...
2. The driver standards committee should have an open discussion of printer limitations.
3. What have been the effects of the latest changes in TEX and what additional effects should be expected.

For those who will remain in town on Wednesday evening, we are planning to have a Dutch-treat outing at a local Country and Western dance hall which is like *Gilley's* that was immortalized in the movie *Urban Cowboy*.

College Station and Bryan are adjoining cities. The university and the hotels for the conference are at the border between the cities. We are about 90 miles northwest of Houston's Intercontinental Airport, 110 miles from Houston's Hobby Airport, 170 miles south of Dallas–Ft. Worth airport, and 100 miles east of Austin's airport. NASA-Houston is 15 miles south of Houston's Hobby Airport. San Antonio is about 170 miles away. We are served by commuter airlines from Dallas (American and Delta) and Houston (Continental). We are on the Greyhound Bus route and Amtrak routes between Dallas and Houston. We have public transportation but it does not run with the frequency that some would like.

We will make every effort to have a wonderful conference and I hope we have a large attendance. Our weather will not be as good as Stanford's, and we may not be on the main line, but we are aggressive and happy TEXers. We feel you will have a wonderful time. The annual meeting brochure has information on how to get more info about vacations in the area, and there are many fun ones to be had. (Contact the TUG office if you have not received a copy of this brochure.)

◇ Bart Childs
Dept of Computer Science
Texas A&M University
College Station, TX 77843-3112
bart@cssun.tamu.edu

Donald E. Knuth Scholarship

Frank Mittelbach was honored at the Tenth Annual Meeting, Stanford University, as the 1989 Knuth Scholarship Winner. His article "An environment for multicolumn output" (*TUGboat* 10, no. 3) was the basis for his selection. Frank has volunteered to serve on the 1990 selection committee.

We are pleased to announce the Fifth Annual "Donald E. Knuth Scholarship" competition. The award is an all-expense-paid trip to either 1990 meeting and one of the Short Courses immediately following the meeting selected. This year's meetings are at Texas A&M University (June 18–22) and the University College Cork (Ireland) (September 10–15). The competition is open to all 1990 TUG members holding support positions that are secretarial, clerical or editorial in nature.

To enter the competition, applicants should submit to the Scholarship Committee by April 30, 1990, the input file and final TEX output of a project that displays originality, knowledge of TeX, and good TeXnique. The project may make use of a macro package, either a public one such as LaTeX or one that has been developed locally; such a macro package should be identified clearly. Such features as sophisticated use of math mode, of macros that require more than "filling in the blanks", or creation and use of new macros will be taken as illustrations of the applicant's knowledge. Along with the TeX files, each applicant should submit a letter stating his/her job title, with a brief description of duties and responsibilities, and affirming that he/she will be able to attend the meeting of his/her choice, specifying meeting and course preference.

Selection of the scholarship recipient will be based on the TeX sample. Judging will take place April 30 - May 20, and the winner will be notified by mail after May 20.

All applications should be submitted to the Scholarship Committee at the appropriate address:

Submissions from North America:

TeX Users Group
Attn: Knuth Scholarship Competition
658 North Main St
P. O. Box 9506
Providence, RI 02940
email: `TUG@math.ams.com`

Submissions from other countries:

Frank Mittelbach
Electronic Data Systems GmbH
Eisenstraße 56
D-6090 Rüsselsheim
Fed Republic of Germany
email: `pzf5hz@drueds2.bitnet`

Coordination of Non-English Use of TeX

Michael J. Ferguson

As indicated in Barbara Beeton's column in *TUGboat* 10, no. 3, I have volunteered to collect hyphenation patterns for "all" languages. In fact, I have volunteered to act as a coordinator of the non-English use of TeX. This includes the collection of patterns, special macros, special LaTeX styles, fonts, lists of inadequacies, and lists of things that TeX does better than any, or at least most, other systems. Since we are entering a new era, with TeX 2.993 ... to become TeX 3.0 in the near future, all of the information to be distributed will be compatible with the new version. I am also especially interested in the rationale for the various national typesetting conventions, including, of course, hyphenation. I would like to eventually create a compendium for the various languages and national groups. An interesting example of an unresolved multilingual problem is whether a phrase in a second language should be hyphenated according to the rules of a that language or of the language of the reader, most likely the first language.

The exact form of the distribution of the special language-dependent information is still to be determined. I expect to include a referral service to the principals in the various countries that have supplied the information. Along with patterns for French that are part of the current Multilingual TeX distribution, I have received additional patterns for Danish, Dutch, German, Icelandic, Portugese, Russian, and Swedish. For Icelandic, Jorgen Pind also supplied a set of fonts. Both Danish and Swedish patterns should be easily adaptable to the new TeX. The Russian patterns assume Cyrillic fonts that are apparently still in development. It is a good beginning but we still have a long way to go.

Since the new TeX is fully 8-bit, the patterns should include all the national characters that are needed for typesetting. Unfortunately, the TeX community has not agreed on a standard way of encoding these characters. For pattern and information interchange, the currently most universal encoding in the TeX community is TeX's backslash encoding. Thus a " é " would be encoded `\'e` and a " Å " would be `\AA`. Some simple macros can be written that will allow TeX's `\pattern` primitive to accept these sequences and replace them with a single 8-bit character. To exploit the new TeX, we need to have all the national characters, at least for the "Latin" European langauges, in (hopefully) a single integrated font. Thus we need fully 8-bit fonts as soon as possible. When this is accomplished, the need for national fonts should disappear. Along with the letter forms, an agreement on the 8-bit code is needed. I would suggest a basis for that encoding could be the ISO-Latin 1 encoding scheme. This encoding is virtually identical to the Digital (DEC) Multinational character codes. ISO also has a number of other encoding schemes for other character sets. We should give strong consideration to following these where possible.

Unfortunately, until the new fonts arrive, it will be very difficult to use the new TeX in anything other than the old way. However, it is possible to test patterns for the new TEX using the current Multilingual TEX. Patterns compatible with Multilingual TEX can be used, almost untouched, in the new TEX. I have done so with the current French patterns, but needed to make a small modification to the new TEX to do so. The basic idea of the modification is to use, internally, the 8-bit equivalent of a character such as " î " which is *ee* in hex, and to replace it with the explicitly accented character just before it is sent to the `.dvi` file. Although currently the modification allows only two letter sequences such as `'e` and uses TEX's standard accent algorithm for the character rebuild, the idea is quite powerful and could be easily enhanced if necessary. Since I consider it really only an interim "fix" for the new TEX, I doubt that this will be either necessary or desirable. This modification has allowed me to use the new TEX with the current fonts. I firmly believe that it will permit us to "root out and destroy the obsolete 7-bit systems" by allowing us to work while waiting for the new fonts. The modification is quite benign and should be easily incorporated in any port.

As part of the coordination function of the non-English use of TEX, I have also agreed to act as coordination point for the Philological Group, originally organized by Reinhard Wonnenberger, and most recently led by Dominik Wujastyk. I would hope to obtain many important suggestions from this group.

Please send contributions, and requests for the "character substitution" change file to me at

`mike@inrs-telecom.uquebec.ca`

◇ Michael J. Ferguson
INRS - Télécommunications
Université du Québec
3 Place du Commerce
Verdun H3E 1H6
Québec, Canada
`mike@inrs-telecom.uquebec.ca`

Using TEX 3 in a Multilingual Environment — Some Ideas

Peter Breitenlohner

As a physicist at a research institute I have been using TEX (2.x) in a bilingual (german + english) environment for many years. In the past we had for each format package (`plain`, LATEX, AMS-TEX, ...) two versions, one with english and one with german hyphenation patterns. With the new TEX 3 this will probably change: we will use versions of the format packages containing hyphenation patterns for both languages and the user will have to select the appropriate language (globally at the beginning of his input and maybe locally if his document contains parts from the other language).

This seems to be the right time to consider how the new multilingual TEX 3 can be used in a way which is first of all portable, then user friendly, and finally efficient (in that order).

I strongly advocate that in the near future *all* macro packages be rewritten to support multilingual environments in a uniform way.

In the following I will try to discuss what should and what can be done for such a situation.

1. Language specific aspects of TEX

1.1. Assigning numbers to the languages. Recently there have been various proposals (e.g. in TEXhax) how to assign numbers to the various languages. In my opinion it is completely unnecessary to universally assign fixed numbers to languages. Numbers should be assigned to languages by a `\newlanguage` macro in the same way as box register values are assigned by `\newbox` and should be referred to by their symbolic names. When I first proposed to Don Knuth that such a `\newlanguage` macro should become part of `plain.tex` he was very reluctant, but since then he has included a `\newlanguage` macro in the latest version of `plain.tex`.

In a multilingual environment one may or may not want to include the english patterns, and in a monolingual non-english version of TEX the english patterns should certainly be excluded. In order not to change `plain.tex` one should therefore have a dummy file `hyphen.tex`, and the original (english) `hyphen.tex` could be renamed `hyphen.eng` (or similar).

Thus as absolute minimum one could specify the following `INITEX` job:

```
\input plain
\newlanguage\english \language\english
  \input hyphen.eng
\newlanguage\german \language\german
  \input hyphen.grm
\dump
```

The user can then, in his TEX job, select his language with the command \language\english.

1.2. Setting up an environment for each language. From past experience we know that more needs to be done. For german texts one would probably want a larger value for \tolerance than the value (200) chosen in plain.tex for english texts. Other languages might want to change the values for \lefthyphenmin and/or \righthyphenmin or similar.

In addition one would want to make " an active character for german and define it such that "a, "s, "ck, "ll, etc., expand to something like \"a, \ss, \discretionary{k-}{}{c}k, l\discretionary{l-}{}{}l and so on. (The actual definitions used for german are different, but this is the basic idea.)

1.3. Language specific texts. Macro packages designed for multiple languages should support the possibility of selecting texts, e.g., for headings, in a way that depends on the language currently selected. (I strongly advocate that all format packages should be redesigned in this sense in the near future.)

Here we find a further complication. Both german and austrian (the german language as it is used in Austria) use the same patterns for hyphenation but they use different names e.g., for the month January: Januar vs. Jänner. Technically I would like to distinguish languages using different patterns and within each language different dialects which differ in their language specific texts to a lesser or larger extent.

2. The problem

In the past such things have often been put on top of the macro package (e.g., german.sty on top of latex.tex). With TEX 3 and its capability to handle multiple patterns it seems natural that the basic mechanisms for language dependencies should be defined when the hypenation patterns are digested, i.e., immediately after plain.tex (or lplain.tex) but before further macro definitions (latex.tex, amstex.tex or whatever).

There are (at least) three different aspects of the language dependencies in TEX:

2.1. The installation. At a particular installation (i.e. computer) there will exist for each macro package one or more .fmt files, each one supporting one or more languages and/or dialects. On a mainframe computer one will probably try to support all languages in one .fmt file; on a PC with its limited memory one might prefer a different .fmt file for each language and/or dialect.

2.2. The author of a macro package. The author of a macro package should define all texts through control sequences which expand to a text that depends on the language currently chosen by the user.

2.3. The user. At the start of his input the user has to specify the language of his text (among the languages and/or dialects supported by the .fmt file). If his documents contain parts in a different language he might want to locally change the current language.

3. Proposal for a design

First of all, the technical difference between language and dialect should be of no concern to the user or to the author of a macro package. Moreover the decisions made by the installation (choice of supported languages) should be decoupled from those made by the author (definition of language dependent texts) and by the user (selection of the current language).

3.1. The installation. Immediately after reading plain.tex (or lplain.tex) but before reading further macro definitions, the installation should define the supported languages and/or dialects. This could be done with macros \deflanguage (which uses \newlanguage) and \defdialect as follows:

```
% define and select \english
\deflanguage\english{eng}{eng_setup}%
  {eng_reset}
\input hyphen.eng
% define and select \german
\deflanguage\german{grm}{grm_setup}%
  {grm_reset}
\input hyphen.grm
% define and select \austrian
\defdialect\austrian{...}{...}
  ...
```

where the first four would probably be contained in a file language.eng, the next six in a file language.grm, etc., i.e. the actual input would probably be

```
% definitions for languages
% (\deflanguage etc.)
\input language
% definitions for english
\input language.eng
% definitions for german (and austrian)
\input language.grm
 ...
```

The macro `\deflanguage` would first of all assign language numbers to `\english` and `\german` (via `\newlanguage`) and would define macros `\s@tenglish`, `\res@tenglish`, `\s@tgerman` and `\res@tgerman` to be invoked by `\setlanguage` in a suitable way. `\setlanguage\german` (or `\setlanguage{german}`) would invoke `\s@tgerman` which includes in its definition the `grm_setup` code which would make " an active character and add it to the `\dospecials` list (needed e.g., for `\verbatim`), would remember the current value of `\tolerance`, set a new value for `\tolerance` and set `\language=\german`. All this would remain in effect until either the current group ends or a different language is selected through `\setlanguage` which would first invoke `\res@tgerman` which includes the `grm_reset` code to cancel the effect of the `grm_setup` code. Similarly `\setlanguage \austrian` would invoke first `\s@tgerman` and then `\s@taustrian`, set `\language=\german` and `\chardef\dialect=\austrian`.

The current values of `\language` (and `\dialect`) will be used in the expansion of the macros for language dependent texts. Here the idea is that a dialect uses the texts for its language unless a different text is specified explicitly.

Since the difference between language and dialect should be irrelevant to both user and author it must be equally possible to specify that one language uses the texts of another language as defaults. This might, e.g., be important if at some future time a dialect is promoted to a language (with its own hyphenation patterns).

The names for languages and/or dialects (as well as the special setup for each of them) should be coordinated by TUG and/or the national TEX users groups.

Note added in Proof: It would be convenient to assign unique ASCII codes in the range 128–255 to all the special characters used by various non-english languages (compare Yannis Haralambous in *TUGboat* 10, no. 3 (1989), pp. 342–345). These characters should, however, be created through suitably defined language specific active characters (such as the german ") and not as ligatures. In addition one should use standardized virtual fonts (see Don Knuth in this issue of *TUGboat*, p. 13) to construct them from characters present in the standard Computer Modern fonts.

3.2. The user. The user simply specifies `\setlanguage\english` or `\setlanguage \austrian` which either does everything required for the language and/or dialect desired or, if the desired language is not supported by the `.fmt` file, gives an error message.

3.3. The author of a macro package. First of all the author of a macro package must define all language dependent texts through control sequences. These control sequences should be defined such that their expansion (technically expansion of expandable tokens and nothing else!) yields the desired text depending on the current values of `\language` (and `\dialect`).

There is, however, a problem. Defining, e.g.,

```
\def\m@nthjan{%
  \ifnum\language=\english January\else
  \ifnum\language=\german
   \ifnum\dialect=\austrian J"anner\else
    Januar\fi\else
  \ifnum\language=\italian Gennaio\else
      ...\fi\fi\fi}
```

is no solution for several reasons. First this leads to an error if any of the language/dialect names: `\english`, `\german`, `\austrian`, `\italian`, ... is undefined (although this could be avoided somehow). Next this would require updating these definitions each time a new language is added. Finally, for an installation which uses just one language, such a definition wastes token memory (and time).

Here I would like to propose a different scheme. Somewhere in his file (probably at the beginning or end) the author of the macro package lists all the macros which need to be defined in a language dependent way (`\m@nthjan` of the example above and all the other ones) and maybe supplies default values in case no language specific values are found. To be specific let us assume all this happens in the file '`abcdef.tex`'. Then the author would use a special macro `\makelanguage` to define the language dependent texts:

```
\makelanguage{abcdef}%
  {\m@nthjan{...}...\m@nthdec{...}...}
```

which inputs a file `abcdef.eng` (specifying the english texts) if and only if english is among the supported languages, a file `abcdef.grm` (specifying the german texts) if and only if german is among the supported languages, and so on. Through this process it would be possible to end up with macro definitions for `\m@nthjan` etc. which are tailored to the set of languages supported by the `.fmt` file.

Clearly all this requires some rather complicated macros to do the job. In addition these macros would not be terribly fast but they would be invoked only once for each set of macros with language dependent text. Here I think portability and flexibility are more important than efficiency! The suffixes `.eng` for english or `.grm` for german are supplied as an argument of `\deflanguage`. Unfortunately these suffixes should be restricted to three characters because some systems (DOS and VMS) allow only that much, otherwise the full names (e.g., `english`) would be preferable.

The precise form of the macro definitions manufactured by `\makelanguage` should be of little interest to author or user. They are somewhat analogous to the `PASCAL` and `TEX` code produced by the `WEB` system programs `TANGLE` and `WEAVE`. These macro definitions should, however, be optimized whenever several languages use the same text. In particular, if all languages use the same text (or there is only one language defined) the replacement text for the macro should be simply this text.

Sometimes a user wants a different text (for one or several languages) than what is supplied by a macro package with its files for language dependent texts (`abcdef.eng` and `abcdef.grm` in the example above). Without precise knowledge of the macro definitions constructed by `\makelanguage` this requires a macro `\changelanguage` which could be used, e.g., in the form

```
\changelanguage
  \somemacro{\somelanguage{...}...}
```

to change the replacement texts of `\somemacro` for the languages `\somelanguage, ...` .

4. Protection

Some of the macro definitions discussed above, in particular `\setlanguage`, certainly must be protected against expansion if they are, e.g., written to an external file. Here I would, however, propose a slight deviation from LaTeX's scheme. `\setlanguage` should be defined via

```
\global\defprotect\setlanguage\setl@ng
```

with the definitions

```
\def\defprotect#1#2{\def#1{\pr@tect#1#2}}
\def\doprotect{\let\pr@tect\d@protect}
\def\noprotect{\let\pr@tect\n@protect}
\def\n@protect#1{}
\def\d@protect#1#2{\noexpand#1}
\noprotect % normally, no protection
```

Thus the sequence

```
{\doprotect
 \immediate\write{\setlanguage}}
```

would write the string '`\setlanguage`' (not '`\setl@ng`') which would still be protected when read in and written again.

5. Summary

In the preceding I have discussed the design of a scheme to handle multiple languages in TeX 3. I have intentionally left out almost all details of how such a scheme can be implemented. (At present a preliminary version of all the required macros is being tested.) For the moment it seems more urgent to agree on a design (including the user interface) than on details of how such a design can be realized through macro definitions.

◇ Peter Breitenlohner
Max-Planck-Institut für Physik
München
Bitnet: `PEB@DMOMPI11`

Software

Erratum: The New Versions of TeX and METAFONT TUGboat Vol. 10, No. 3

Donald E. Knuth

Editor's note: The following should replace the second full paragraph on page 326, column 1:

> The special whatsit nodes are inserted automatically in unrestricted horizontal mode (i.e., when you are creating a paragraph, but not when you are specifying the contents of an hbox). You can insert a special whatsit yourself in restricted horizontal mode by saying `\setlanguage`⟨number⟩. This is needed only if you are doing something tricky, like unboxing some contribution to a paragraph.