Software

Editor's Introduction

Helmut Jürgensen

The Software Column of this TUGboat issue is a call for help—your contributions or suggestions!

Some of the things we would like to receive are outlined below. If you consider submitting material which you feel would fit into the Software Column but which is not covered by the list, please submit it anyway—or contact me.

TEX has come of age now—everybody said so at the TUG meeting. The main interest of TUG members seems to shift away from "mere" implementation questions gradually, and towards "real" typesetting problems. Nevertheless, it also seems that there are still quite a few difficulties arising when embedding the TEX and **METAFONT** systems into different environments in a user-friendly fashion. A lot of software has been and is being written to handle the problems. In fact, judging from some discussions I had at the TUG '84 meeting, it is written again and again and again. Therefore, I think the TUGboat and this column could help in distributing information and discussing ideas.

Please send descriptions of TEX and **META-FONT**-related software: They could be short, giving only name, purpose, availability, and—enough—important technical data. They could be long as well, if of sufficiently general interest. In some cases novel software methods may require a more thorough treatment. WEB-related problems would form another area of interest.

In general, information about software used in the implementation of the TEX, **METAFONT**, and WEB systems, in the adaptation to your environment or your research work, about enhancements to these systems, or software engineering comments related to these systems are welcome.

I should also appreciate receiving suggestions about the column in general, comments on articles, hints about related publications elsewhere, etc.

Please send your contributions and ideas to

Helmut Jurgensen
Department of Computer Science
The University of Western Ontario
London, Ontario
Canada N6A 5B7

You can also reach me by phone (519-679-3039) or via usenet (deepthot!uwo!uwo-hobbit!jurgensen).

Please remember, I need your contributions—otherwise the column will starve—and I must have them no later than two weeks before the general TUGboat deadline.

TEXtensions

How to Run TEX in a French Environment: Hyphenation, Fonts, Typography

Jacques Désarménien

1. Introduction

One of the greatest achievements of text-processing programs—especially TEX—is that they allow anybody to produce beautiful, typeset-like documents by merely typing a correctly prepared manuscript on a terminal, using a keyboard very similar to that of an ordinary typewriter.

As a consequence, before trying to train professional typists in TEX, one must ensure that a replica of the keyboard they are used to will appear on the terminal. In our situation, this means that some extra characters—é, è, ç among others—have to be included where they usually appear on a French keyboard. Some others must be displaced: for example, a and q have to be permuted and the digits retain their positions but must be typed by using the SHIFT key. The solution to these specific problems, and a certain number of improvements to the input of TEX, have been given by D. Foata and Y. Roy in Strasbourg [fr].

The main concern in processing French text is hyphenation. The set of patterns that TEX uses to hyphenate English works very poorly in French. The procedures are fundamentally different, not only because the etymology has much more importance in English, but also because the syllabifications do not obey the same rules in both languages. The same word might not be hyphenated the same way in French and in English: `ma-gni-fi-cence` *vs.* `mag-nif-i-cence`. We explain in another article

[d] how we have produced a set of patterns suitable for French. In section 3 below, we give a short account of this process, and we discuss the technical aspects of loading these patterns into TeX.

As we will see, in order to realize an acceptable hyphenation of French using TeX, the fonts have to be modified. This gives the opportunity to improve the appearance of some of the accented characters. Unfortunately, due to some restrictions of both **METAFONT** and TeX, it was impossible to fully achieve this last goal. We explain why, and propose some remedies.

Finally, some typographical conventions are different: Spacings are different in French and the quotation marks are replaced by so-called 'guillemets'. These characteristics have been included in our 'French' fonts and our 'French PLAIN' format.

Even if the ideas we develop were specifically oriented towards the purpose of using TeX to typeset French, we are certain that most of them are still valid for other languages.

2. French fonts

The necessity of having special fonts comes primarily from the way TeX hyphenates [k]. Let us just say that TeX attempts no hyphenation after the third letter preceding a non-letter or 'explicit kern', where TeX considers a letter to be a character with category code 11 or 12, having a non-zero lower-case code (\lccode).

PLAIN TeX produces the accented characters by superposing an accent over a letter. This is achieved by means of explicit kerns. It follows that no hyphenation is possible in a word containing accents, after the third letter preceding the first accent. Unfortunately many French words contain accented characters, the most frequent being the é: out of the 49,962 entries of our on-line French vocabulary [g], 15,970 contain at least one accented letter and among them 13,477 contain at least one é.

Another problem arises when typesetting accented characters with TeX. Many times, the placement of the accent is deficient. On low-resolution printers, it can even be inconsistent. Even if the rounding and conversion of the DVI units to the raster units is improved—that eliminates the worst defects and inconsistencies—some accents still have to be adjusted. Over some lower-case letters, they have to be slightly shifted, and they are too high and too steep over the capitals. Of course, some of these defects can be corrected by inserting extra

kern, or even by rewriting the macros for the accents. Nevertheless, the best solution is to include all the accented characters as entities in the font: It gives the best positioning of the accents and enables TeX to hyphenate.

French uses thirteen accented characters: â, ê, î, ô, û, é, à, è, ù, ë, ï, ü and ç, plus the ligature œ. Even if the corresponding upper-case characters are missing from 'French' fonts as distributed by some type foundries, at least the accented upper-case E's and the C cedilla cannot be dispensed with. In fact, good typography requires all fourteen upper-case special characters [frey].

TeX can handle 'extended' fonts of up to 256 characters, but only the first 128 are defined in the fonts currently used, leaving a lot of room available for extra characters. The solution seems obvious: Place the extra characters in the second half, with character codes from 128 to 257. Unfortunately, this is unfeasible at the moment. The main reason is that only characters with codes from 0 to 127 are allowed to have a nonzero \lccode. One further restriction comes from the fact that the 'old' **METAFONT** cannot generate fonts of more than 128 characters. This last difficulty will be eliminated when the 'new' **METAFONT** is fully operational.

Consequently, we tried to devise a temporary solution, allowing us, at least, to build and try the hyphenation patterns. This made it necessary to include a minimal number of accented characters.

By comparing the Computer Modern font tables and the standard ASCII character codes, it appears that only the codes from 0 to 32 and 127 are fully available. (Some others could be used, except in the 'typewriter' fonts.) As it was impossible to move all these characters to positions over 127, we had to get rid of a certain number of them. The best—if not the only—candidates were the unslanted Greek capitals, occupying positions 0–10. Not even the accents can be removed: The acute is needed on any vowel if a quotation from Spanish is to be typeset, and the same is true with the grave accent in Italian, the umlaut accent in German, and the circumflex in Esperanto.

Therefore, we had to abandon some of the accented characters. The ü is never used in modern French, except in a few proper names. The ù is used in one word—où (where)—and à in a few words, all of which are too short to be hyphenated. Each of the others may occur in the middle of words (though very seldom will ë and ï). We decided to include them. They are the ten accented letters: â,

ê, î, ô, û, é, è, ë, ï and ç. The ligatures œ and Œ already appear in the 'Computer Modern' fonts.

As it is safer for a lower-case letter to be equal to its \lccode—this last parameter is an essential element of the \lowercase primitive—the code 0 cannot be used for a lower-case letter. It could be possible to assign this code to the É, which appears quite often at the beginning of a word. But the lower-case é could not be converted to the upper-case É via \uppercase: a character with 0 as its \uccode remains unchanged. So we did not retain this possibility, hoping that, very soon, new fonts and/or a change to the restrictions about the \lccode will allow more flexible capabilities of adding extra letters.

We also removed the Spanish open question and exclamation marks and replaced them by the opening and closing guillemets, for which we wrote some lines of (old) **METAFONT** code. Fortunately, these guillemets now have the same character codes as the ASCII characters '<' and '>'. Moreover, the guillemets usually are not available in typewriter style, and so do not conflict with the characters '<' and '>' in the typewriter fonts.

The accented characters have been produced by adding the **METAFONT** codes for the accent to the code for the letter, and slightly adjusting the horizontal positioning of the former.

Finally, we modified and increased the number of kerning instructions in the **METAFONT** code so that the spacing is consistent: the same implicit kern is inserted between x and é as between x and e, for example. We added the same kern before and after œ as before o and after e, and did the same for Œ. (In our opinion, this could be included in the ordinary fonts.) We noticed that the spacing was wrong between f and î in roman style, and inserted the italic correction. For the same reason some kern had to be specified between the apostrophe and the accented î. The result is acceptable. Unfortunately, no satisfying result could be devised in the case of the sans-serif fonts, especially in such a word as fît (it exists!), where the rhythm of the strokes is important. This confirms the point that the design of the accented characters has to be part of the work of the designer at a relatively early stage. Let us end by mentioning that, for reasons to be explained later, the 'extra space' parameter, known by TEX as \fontdimen7, has been set to a very small negative dimension.

A sample of one of our 'French Computer Modern' fonts can be found in Appendix I.

3. The hyphenation patterns

Let us just give a short account of the way we constructed a set of patterns to be used when typesetting French. This work is more extensively described in another article [d].

Liang produced the set of patterns currently used by TEX for hyphenating English by applying his 'PATGEN' program to a slightly edited version of the *Webster's Pocket Dictionary* (see [l]). The same approach was impossible in French: no French dictionary indicates the feasible break positions. Actually, the rules are quite simple, mostly based on the succession of vowels and consonants, with some unbreakable groups (bl, br, ch, chr, gn, ...). These rules were translated without major difficulties into a first set of 371 patterns, called 'phonetic patterns', from the fact that they were obtained from the phonetic syllabification rules. Unfortunately, in some instances the etymological hyphenation is preferred: for compound words, neologisms and scientific terms. Such a division is needed for about 3.5% of the words, and it appears that most of them are obtained by adding a prefix to an already existing term. The suitable 'etymological patterns' were found by scanning our on-line French vocabulary [g], containing almost 50,000 entries, and seaching for the occurrences of possible patterns. The resulting second set of 403 patterns ensures the right etymological hyphenations where they are needed, and in most nonexistent but possible cases (scientific terms, neologisms). The resulting set of 774 patterns was finally tested by checking once more the French vocabulary.

The conclusion is that 100% of the French words are correctly hyphenated—in our opinion, since no precise rule establishes when the etymological hyphenation has to be preferred to the phonetic one. No permissible hyphenations are missed, and no impermissible ones are introduced. Some words of foreign origin are specified in the \hyphenation list. One of the advantages of the method used to produce the patterns is that the meaning of each of them is clear: They can be removed, modified, and some more can be added. If enough care is taken, the effect of such manipulations will be perfectly predictable. Hence it is easy to adapt the set of patterns to anyone's needs, and to correct the possible errors.

In order to load the patterns, and to ensure a proper hyphenation of the words containing extra

characters, we must assign these characters lower-case codes. These are obviously equal to the character codes. To allow \uppercase to work in an acceptable way, we assign to the accented characters the upper-case codes of the corresponding unaccented capitals. This does not apply to the ligatures œ and Œ: they just exchange their lower-case and upper-case codes like ordinary letters. Finally, the apostrophe is given a \lccode equal to its character code, 39.

The only admissible characters in the patterns are letters and digits, the latter, in our case, serving as coefficients (see [k, l]). In particular, no control sequences are allowed in the list of patterns. Consequently, the extra characters must be specified through the 'double hat' notation, used for specifying the ordinarily 'inaccessible' ASCII characters. For example, the é is denoted by ^^F, corresponding to the code 6. (This kind of obscure encoding is used only when loading the patterns.) In our case, all of the special characters, except ë and ï, but including œ and the apostrophe actually appear in the patterns (^^A to ^^G, ^^I, ^^[and ').

Some of the codes used for accented characters have been assigned special category codes in the PLAIN file: ^^A, the ASCII code 1 (on SAIL the 'downarrow'), can be used for subscripts (category 8), and the ASCII tab, ^^I, is synonymous with the space (category 10). Just before loading the patterns, these category codes have to be changed to 12 ('other character'). The initial values are restored just after loading. Obviously, no 'tab' (temporarily equivalent to è!) may appear in between. For the same reasons, in case some other ASCII characters are given a special meaning (due to a peculiarity of the local system for example), care must be taken to avoid any confusion with special characters.

4. Modifications to the PLAIN format

In order to test our set of hyphenation patterns, and to devise some simple ways of adapting TeX to French typography, we had to make a new format file. What we produced is only a modified version of PLAIN as it is described in [k, Appendix B]. Obviously, a more specific use of TeX in French—as is the case in English—will require its own format. In this case, it would be preferable to make it by incorporating the various French typographical conventions from the beginning instead of adapting an existing format.

First of all, as the unslanted Greek capitals are missing from the 'French' fonts, we decided to keep all the original ones and add some 'French' fonts. Hence, we defined two more families of fonts, the French roman and the French text italic, to begin with, and we contemplate adding new ones if needed (possibly French roman bold). Each of these families consists of only the ten point font, but in the near future, the same sizes will be available for them as for the (ordinary) roman and text italic. Corresponding to the macros \rm and \it we introduced \frm and \fit. We could have redefined \rm and \it, but, because of the lack of Greek capitals, we would have had to modify some code in the 'math macros' section. Furthermore, as long as some sizes are missing in 'French' fonts, it would have been necessary to let PLAIN switch, inside the same family, between 'ordinary' and 'French' fonts. It seemed preferable to let the typist know without ambiguity what kind of font he is currently using. Of course, it is not difficult to include such possibilities in PLAIN, and that will probably be the case when more 'French' fonts are included. After the French PLAIN format has been input, the default font is French roman 10 pt.

A happy consequence of our decision to retain all the fonts used in PLAIN is that no modification is needed to the definition of the math macros. The only modifications, apart from the font loading and some minor change at the very end, appear in the fifth section ('macros for text') of PLAIN. To minimize the changes, we decided to keep all the control sequences for the accents. To avoid the insertion of spaces inside a word, we added only the variants \5 for the cedilla and \e/ and \E/ for the ligatures œ and Œ, ordinarily obtained by \c, \oe and \OE respectively. For the same reason, we decided that \^i and \"i could be used instead of \^\i and \"\i. (We are almost certain that nobody will ever need to accent a dotted i; in case we are wrong, it is still possible to make one by \^{\char'i}.)

To make the same control sequence behave in two different ways according to the nature of the font—'French' or 'standard'—a parameter has to be introduced to test it. The 'extra space' (\fontdimen7) has been set to a very small negative value for the 'French' fonts. This parameter is used when \nonfrenchspacing provides extra space after some punctuation marks. This is not the case in a French context, and in case someone uses it, its value is small enough to make sure that no spacing

will be wrong. It is to be presumed that no font ordinarily needs such a negative extra space.

In math mode, the accent macros will produce the right accented characters, provided the 'French' fonts are used. But the same macros will result in error messages when used for à, ù or any of the accented capitals, or when the font is not a 'French' one. In any case, until new fonts containing all the accented characters are available, the 'French' fonts are intended for typesetting text only.

When fonts other than 'French' fonts are used, or when accented capitals are typed, the French hyphenation is still on, with the restriction we mentioned earlier: no hyphenation will be attempted after the third letter preceding an accent character. This implies that many legitimate breaking points will be missed. This is not a major restriction, since the fonts that do not have French versions are mostly display fonts. And since when using the ordinary \hsize with ten-point fonts, the relatively small number of times that TEX has to hyphenate makes it acceptable.

As we already said, the default font is French roman: the command to set this default appears in the last section ('Hyphenation and everything else') of PLAIN. Thus, any addition to the explicit \hyphenation list can be made at the very beginning of the user's file. Unlike the main pattern list, it may include control sequences, but they must expand into letters. Consequently, the current font must be a French one to allow accented characters to appear in the words for which an explicit hyphenation is given. (The use of the 'double hat' notations would work in any case, but is not recommended because some of the characters have category codes other than 12.) For the same reason, we had to modify the definition of \showhyphens. Instead of testing the hyphenation of a word in \tenrm, it is done in the current font, either French or standard.

Another problem arises from the apostrophe. For TEX, it is a letter. But the same character happens to be used as a single close quote. When it has the latter meaning, it might provoke some bad hyphenations towards the end of the preceding word (the worst examples being something like *co-qs'* or even *co-q"*). Such a hyphenation is very unlikely, both for statistical reasons and because the English-style quotation marks are usually not used in French typesetting since they are replaced by the guillemets.

The last set of macros we added deals with the spacing around some punctuation marks. The semi-colons, exclamation points, and question marks are usually preceded by a thin space, which must not shrink nor stretch. The colons should be preceded by the inter-word space of the current line, which may vary from one line to another. The ordinary inter-word space follows any punctuation. Obviously, no line break is allowed before a punctuation mark. Moreover, these four punctuation marks usually are preceded and followed by a space when typed on a keyboard by French typists. The solution we have adopted is to make these characters active (category code 13). When they occur, they first test the current mode. If in horizontal mode, they remove the last glob of glue, if any (*i.e.*, the space before the punctuation is significant) and in this case replace it with an explicit kern (for the first three marks) or with unbreakable inter-word glue (for the colon). The explicit kern is such that its value is .4 times the inter-word space except in typewriter style, in which case it is exactly the inter-word space.

Finally, there exists a French typesetting convention for which we were unable to devise the correct macros: When a quotation is made *inside* another quotation, it begins with the same mark (*i.e.* opening guillemets). But, until the end of the inner quotation, the opening guillemets are repeated at the beginning of each line and are aligned with the left margin. In a sense, the lengths of the lines are shorter by the width of the opening guillemets during the inner quotation. The difficulty comes from the fact that it is impossible to modify the lengths of the lines inside a paragraph, unless the number of lines to be modified is known in advance.

The French quotes-in-quotes convention appears in the following paragraphs. They could appear in a review of our article [d].

Les problèmes posés par la division étymologique sont ainsi résumés par J. Désarménien (*cf.* [d]) :

« Tout le monde s'accorde à trouver légitime la division : **extra-ordinaire**. Le cas des mots composés est le seul à rencontrer cette unanimité. Dans les autres, les opinions sont très partagées. Frey est le plus catégorique : il n'admet de division selon la formation que pour des mots de composition complètement française, et la rejette sinon ... Le *Code typographique*, tout en adoptant ce même point de vue « ... [reconnaît] néanmoins que certains au- « teurs de travaux scientifiques préfèrent la division

« étymologique qui fait ressortir la racine grecque
« ou latine ». Quant à Gouriou, il écrit : « On pré-
« férera cependant garder la coupure étymologique
« chaque fois que les composants sont *aisément* re-
« connaissables. » (L'italique est de Gouriou.) Telle
est aussi l'opinion de Girodet. »

The solution we give essentially builds a first
list of lines, until reaching the inner quotation; then
it removes the last line from the list and appends
what remains to the page. Then, this last line is
used as the beginning of a second vertical box with
different indentation parameters. Once more, the
last line is removed and the rest is appended to the
page with opening guillemets at the beginning of
each line. Finally, this new last line is appended to
the page as a horizontal list, and it is followed by the
end of the current paragraph. A number of 'dirty
tricks' were used to avoid the use of arguments
inside the macros, to allow the right spacings and
penalties between lines, and to avoid having two
opening guillemets at the beginning of the first line
of the second section when the last line of the first
section is almost \hsize long.

5. Conclusions

Our main goal, when starting this work, was to build
a good set of patterns for French hyphenation. We
think that this has been achieved satisfactorily.
The other problems we had to face—except those
related to the spacing around punctuation marks—
were consequences of this one.

The main limitation arose from the difficulty
of adding characters to the standard fonts. There
are essentially two options for the future. The first
one is to modify the code of TEX itself to allow
characters with codes over 127 to be assigned lower-
and upper-case codes. We did not want to take the
responsibility of a change that could become more
extensive than expected.

The other possibility is to free the first 32
positions of the fonts. The characters currently
appearing in these positions could be moved to codes
over 128: They include Greek capitals, ligatures,
accents and special characters, none of them needing
an \lccode. The opportunity can be taken to
replace the Spanish open exclamation and question
marks by guillemets, which are used in a number
of languages, and it seems essential that they be
accessed directly. The special Spanish marks could
be in the second half of the font. It could also be
possible to design special accents for capitals. In

this manner, a set of 'basic' fonts would be available.
They could be used for typesetting English, even if
some quotations from foreign languages had to be
made.

Then 'local' fonts could be obtained by assign-
ing the first 32 codes to what is needed: in French,
the 13 accented letters plus the ligature œ, in lower
and upper cases, in Spanish the accented vowels, ñ
and ü, and so on. In some cases, characters nor-
mally in the second half of the font would need to
be transferred to the first half, or they could simply
be duplicated: the ligatures œ and Œ in French, the
crossed ø and Ø in Danish and the dotless ı in
Turkish are examples.

One of these choices would make it possible to
limit the changes to the PLAIN format to a few,
well determined areas, such as accents and spacing.
There would be no need to add new fonts. It would
be sufficient to substitute 'local' fonts to the 'basic'
ones. If some care is taken when writing a format,
it would even be possible to process *any* TEX source
with *any* local set of fonts, the best results being
obtained with the right set.

It seems to us that such an approach would
simultaneously facilitate the adaptation of TEX
to a language other than English, and keep its
universality.

There is one last problem one is faced with
when using TEX in a foreign language. It has
been mentioned in the introduction: The input
must fit the local typing habits. Unfortunately
the number of ASCII characters is too small to
allow a direct access to the extra characters. A
solution must involve modified or programmable
keyboards. Obviously, no rule can be established in
this domain. Nevertheless, it seems preferable that
such local systems be used to produce, in a handy
way, a universally understandable TEX file. Such a
system is currently working in Strasbourg.

In addition, I thank Oren Patashnik for his careful reading of the manuscript.

References

[d]   J. DÉSARMÉNIEN, La division par ordinateur des mots français avec le logiciel TEX, preprint.

[fr]   D. FOATA et Y. ROY, A Rationalized French Keyboard for Inputting TEX, Laboratoire de typographie informatique de l'Université de Strasbourg.

[frey] A. FREY, *Manuel nouveau de typographie*, à la librairie encyclopédique de Roret, Paris, 1835.

[g]   M. GROSS, DLAS, vocabulaire français informatisé, Laboratoire d'automatique documentaire et linguistique, Université Paris VII.

[k]   D. KNUTH, *The TEXbook*, Addison–Wesley, Reading, Mass., 1984.

[l]   F. M. LIANG, Word Hy-phen-a-tion by Com-put-er, Ph.D. Thesis, Department of Computer Science, Stanford University, Report No. STAN-CS-83-977, 1983.

Appendix I

Text font 'French roman'

	0	1	2	3	4	5	6	7
'000		â	ê	î	ô	û	é	ç
'010	ë	è	ï	ff	fi	fl	ffi	ffl
'020	ı	ȷ	`	´	ˇ	˘	¯	˚
'030	¸	ß	æ	œ	ø	Æ	Œ	Ø
'040	´	!	"	#	$	%	&	'
'050	(	)	*	+	,	-	.	/
'060	0	1	2	3	4	5	6	7
'070	8	9	:	;	«	=	»	?
'100	@	A	B	C	D	E	F	G
'110	H	I	J	K	L	M	N	O
'120	P	Q	R	S	T	U	V	W
'130	X	Y	Z	[	"	]	^	˙
'140	`	a	b	c	d	e	f	g
'150	h	i	j	k	l	m	n	o
'160	p	q	r	s	t	u	v	w
'170	x	y	z	–	—	˝	~	¨

Appendix II

Modifications to the PLAIN format

Following are the modifications to PLAIN as described in Appendix B of *The TEXbook* [k]. The modified PLAIN format and the patterns given in [d] have been used to typeset reference [d] itself.

In section 4, Font information:

Two new fonts are loaded, and two new families are defined, for French roman and text italic. More fonts in more sizes will complete both families.

```
\font\tenfrm=fmr10
\font\tenfit=fmti10
\newfam\frfam \def\frm{\fam\frfam\tenfrm} \textfont\frfam=\tenfrm
\newfam\fifam \def\fit{\fam\fifam\tenfit} \textfont\fifam=\tenfit
```

In section 5, Macros for text:

The circumflex, acute, grave and dieresis ('trema' or umlaut) accents are entirely redefined. The original macros are removed and replaced by the following.
First, the accented characters are given names.

```
\chardef\@acircon=1
\chardef\@ecircon=2
\chardef\@icircon=3
\chardef\@ocircon=4
\chardef\@ucircon=5
\chardef\@eacute=6
\chardef\@ccedilla=7
\chardef\@etrema=8
\chardef\@egrave=9
\chardef\@itrema=10
```

Then, the superposed versions of these accents are defined. Note the tests for the dotted i for the circumflex and dieresis.

```
\def\@GR#1{{\accent18 #1}}
\def\@AC#1{{\accent19 #1}}
\def\@CIR#1{{\if i#1\accent94 \i \else\accent94 #1\fi}}
\def\@TR#1{{\if i#1\accent127 \i \else\accent127 #1\fi}}
```

Now, the 'built-in' versions of the same accents. Note the tests for the *un*dotted ı in the last two.

```
\def\FRENCH@AC#1{\if e#1\@eacute\else
  \@AC{#1}\fi}
\def\FRENCH@GR#1{\if e#1\@egrave\else
  \@GR{#1}\fi}
\def\@dotlessi{\i}
\def\FRENCH@CIR#1{\def\@param{#1}\if e#1\@ecircon\else
  \if a#1\@acircon\else
  \if i#1\@icircon\else
  \if o#1\@ocircon\else
  \if u#1\@ucircon\else
  \ifx \@dotlessi\@param\@icircon\else
  \@CIR\@param\fi\fi\fi\fi\fi\fi}
\def\FRENCH@TR#1{\def\@param{#1}\if e#1\@etrema\else
  \if i#1\@itrema\else
  \ifx \@dotlessi\@param\@itrema\else
  \@TR\@param\fi\fi\fi}
```

The next one is for the cedilla. Unlike the other French accents, the original macro \c is retained.

```
\def\FRENCH@CED#1{\if c#1\@ccedilla\else
  \c{#1}\fi}
```

Then the final macros are defined, testing the positivity of the 'extra space' parameter of the current font. The control sequence \5 produces the cedilla. The \c still works, but it produces ç by superposing instead of by using the character c cedilla, thus preventing hyphenation.

```
\def\^#1{\relax\ifdim\fontdimen7\font<\z@
  \FRENCH@CIR{#1}\else\@CIR{#1}\fi}
\let\^^D=\^
\def\'#1{\relax\ifdim\fontdimen7\font<\z@
  \FRENCH@AC{#1}\else\@AC{#1}\fi}
\def\`#1{\relax\ifdim\fontdimen7\font<\z@
  \FRENCH@GR{#1}\else\@GR{#1}\fi}
\def\"#1{\relax\ifdim\fontdimen7\font<\z@
  \FRENCH@TR{#1}\else\@TR{#1}\fi}
\def\:{\relax\ifdim\fontdimen7\font<\z@
  \@ccedilla\else \c c\fi}
\def\5#1{\relax\ifdim\fontdimen7\font<z@
  \FRENCH@CED{#1}\else\c{#1}\fi}
\def\e/{\oe}
\def\E/{\OE}
```

The following macros are added to the format. The first series is for the spacing before some punctuations. The punctuations behave in a special way only if they are used in horizontal mode and preceded by a nonnegative amount of glue (a space, for example). Otherwise they append themselves, with category code 12, as usual.

```
\frenchspacing
\catcode'\;=\active
\catcode'\:=\active
\catcode'\!=\active
\catcode'\?=\active
\def;{\relax\ifhmode\ifdim\lastskip>\z@
  \unskip\kern\fontdimen2\font \kern-1.2\fontdimen3\font
  \fi\fi\string;}
\def:{\relax\ifhmode\ifdim\lastskip>\z@
  \unskip\nobreak\ \fi\fi\string:}
\def!{\relax\ifhmode\ifdim\lastskip>\z@
  \unskip\kern\fontdimen2\font
  \kern-1.2\fontdimen3\font\fi\fi\string!}
\def?{\relax\ifhmode\ifdim\lastskip>\z@
  \unskip\kern\fontdimen2\font
  \kern-1.2\fontdimen3\font\fi\fi\string?}
```

The second series deals with inner quotations. It is quite tricky. The macros are used in the following way: The paragraph containing the inner quotation begins with \qquotes; the guillemets opening the inner quotation are typed '*<', the guillemets closing it are typed '>*', then the paragraph ends normally. If there are more than one inner quotation, the final '*' of all but the last one must be replaced by \finniveau2. The example on page 95has been typed:

```
\qquotes <Tout le monde s'accorde ...
... ce m\^eme point de vue *<\ldots [reconna\^it] n\'eanmoins ...
... grecque ou latine>\finniveau2. Quant \'a Gouriou, il \'ecrit :
*<On pr\'ef\'erera cependant ...
... {\fit ais\'ement} reconnaissables.>* (L'italique est ...
... l'opinion de Girodet>\par
```

The opening '<' is mandatory. The closing '>' may be omitted. In that case, no closing guillemets appear at the end of the double quotation. The text preceding '*<' or following the second '*' may be empty. This allows the double quotation to run for more than one paragraph. The paragraphs in between begin with opening guillemets, but end with no special mark, as is customary in French. More precisely,

```
\qquotes (text 1) *<(text 2)*\par
\qquotes *<(text 3)>* (text 4)\par
```

produces a paragraph break in the middle of the double quotation consisting of (text 2)(text 3). Because many groupings are involved in the following macros, any change of parameters must be declared *before* \qquotes in order to be effective inside. Finally the '*' is given a special meaning between \qquotes and the end of the double quotation. But its previous meaning is restored from this point, even if it had been made active and given a special definition before \qquotes.

```
\def\@setpartozero{\widowpenalty=\z@ \clubpenalty=\z@
  \interlinepenalty=\z@ \brokenpenalty=\z@ \displaywidowpenalty=\z@}
\newbox\@tempbi\newbox\@tempbii\newbox\@tempbiii
\newbox\@tempbiiii\newbox\@tempbvi\newbox\@tempbvii
\newbox\@openquotes
\newdimen\@hminusem
\newif\ifnonvoid
\catcode'\*=\active
\def\@transfervbox#1#2{\nonvoidtrue
  \loop
  \setbox\@tempbi=\vbox{\unvbox#1\global\setbox\@tempbiii=\lastbox
    \unskip}%
  \ifvoid\@tempbiii\nonvoidfalse\fi
  \ifnonvoid
  \setbox\@tempbii=\vbox{\unvbox#2\box\@tempbiii}%
  \setbox#1=\box\@tempbi\setbox#2=\box\@tempbii
  \repeat}
\def\@transferaddvbox#1#2{\nonvoidtrue
  \setbox\@tempbi=\vbox{\unvbox#1\global\setbox\@tempbiii=\lastbox
    \unskip}%
  \setbox#2=\vbox{\box\@tempbiii}%
  \setbox#1=\box\@tempbi
  \loop
  \setbox\@tempbi=\vbox{\unvbox#1\global\setbox\@tempbiii=\lastbox
    \unskip}%
  \ifvoid\@tempbiii\nonvoidfalse\setbox#1=\box\@tempbi\fi
  \ifnonvoid
  \setbox\@tempbii=\vbox{\unvbox#2%
    \hbox to\hsize{\copy\@openquotes\unhbox\@tempbiii}}%
  \setbox#1=\box\@tempbi\setbox#2=\box\@tempbii
  \repeat}
\def\@sendtopage#1{\nonvoidtrue
  \loop
  \setbox\@tempbi=\vbox{\unvbox#1\global\setbox\@tempbiii=\lastbox
    \unskip}%
  \ifvoid\@tempbiii\nonvoidfalse\setbox#1=\box\@tempbi\fi
  \ifnonvoid
  \unhbox\@tempbiii\unskip\break
  \setbox#1=\box\@tempbi
  \repeat}
\def\@star{\egroup
  \@transfervbox\@tempbvi\@tempbvii
  \@transferaddvbox\@tempbvii\@tempbvi
  \setbox\@tempbvii=\vbox{\unvbox\@tempbvi
    \global\setbox\@tempbiiii=\lastbox\unskip}%
  \@transfervbox\@tempbvii\@tempbvi
  \noindent \@sendtopage\@tempbvi
  \unhbox\@tempbiiii\unskip\unskip\unpenalty}
```

```
\def\finniveau2{\egroup
  \@transfervbox\@tempbvi\@tempbvii
  \@transferaddvbox\@tempbvii\@tempbvi
  \setbox\@tempbvii=\vbox{\unvbox\@tempbvi
    \global\setbox\@tempbiiii=\lastbox\unskip}%
  \@transfervbox\@tempbvii\@tempbvi
  \noindent \@sendtopage\@tempbvi
  \setbox\@tempbvii=\vbox\bgroup\@setpartozero
  \catcode`\*=\active \let*=\@staropen
  \noindent \unhbox\@tempbiiii\unskip\unskip\unpenalty}
\def\@staropen<{\global\setbox\@openquotes=\hbox{<}\egroup
  \setbox\@tempbvi=\vbox{\unvbox\@tempbvii
    \global\setbox\@tempbiiii=\lastbox\unskip}%
  \@transfervbox\@tempbvi\@tempbvii
  \noindent\@sendtopage\@tempbvii
  \setbox\@tempbvi=\vbox\bgroup\@setpartozero
  \catcode`\*=\active
  \let*=\@star
  \hangindent=\wd\@openquotes\hangafter=1
  \setbox\@tempbvii=\hbox{\unhcopy\@tempbiiii\unskip\unskip
    \unpenalty}%
  \@hminusem=\hsize \advance\@hminusem by -2em%
  \ifvoid\@tempbiiii\indent\copy\@openquotes
  \else
  \parindent=\z@
  \ifdim \wd\@tempbvii>\@hminusem \unhbox\@tempbvii\break
  \else \unhbox\@tempbvii\nobreak\ \copy\@openquotes\fi\fi}
\def\qquotes{\setbox\@tempbvii=\vbox\bgroup\@setpartozero
  \catcode`\*=\active \let*=\@staropen}
\catcode`\*=12
```

In section 8, Hyphenation and everything else:

The \showhyphens macro is modified, and the default font is French roman 10 pt.

```
\def\showhyphens#1{\setbox0\vbox{\parfillskip\z@skip\hsize\maxdimen
    \pretolerance\m@ne\tolerance\m@ne\hbadness0\showboxdepth0\ #1}}
\normalbaselines\frm % select french roman font
```

The set of patterns and exceptions is different, and appears *after* the font is set to French roman to allow accents to appear in the hyphenation list.

```
\input frhyph
```