

1 The Catalan language

The file `catalan.dtx`¹ defines all the language-specific macro's for the Catalan language.

For this language only the double quote character (") is made active by default. In table 1 an overview is given of the new macros defined and the new meanings of ". Additionally to that, the user can explicitly activate the acute accent or apostrophe (') and/or the grave accent (‘) characters by using the `activeacute` and `activegrave` options. In that case, the definitions shown in table 2 also become available².

<code>\l.l</code>	geminated-l digraph (similar to l·l). <code>\L.L</code> produces the uppercase version.
<code>\lgem</code>	geminated-l digraph (similar to l·l). <code>\Lgem</code> produces the uppercase version.
<code>\up</code>	Macro to help typing raised ordinals, like 1 ^{er} . Takes one argument.
<code>\-</code>	like the old <code>\-</code> , but allowing hyphenation in the rest of the word.
<code>"i</code>	i with diaeresis, allowing hyphenation in the rest of the word. Valid for the following vowels: i, u (both lowercase and uppercase).
<code>"c</code>	c-cedilla (ç). Valid for both uppercase and lowercase c.
<code>"l</code>	geminated-l digraph (similar to l·l). Valid for both uppercase and lowercase l.
<code>"<</code>	French left double quotes (similar to <<).
<code>"></code>	French right double quotes (similar to >>).
<code>"-</code>	explicit hyphen sign, allowing hyphenation in the rest of the word.
<code>" </code>	disable ligature at this position.

Table 1: Extra definitions made by file `catalan.ldf` (activated by default)

These active accents characters behave according to their original definitions if not followed by one of the characters indicated in that table.

The macro `\LdfInit` takes care of preventing that this file is loaded more than once, checking the category code of the @ sign, etc.

```
1 {*code}
2 \LdfInit{catalan}\captionscatalan
```

When this file is read as an option, i.e. by the `\usepackage` command, `catalan` could be an ‘unknown’ language in which case we have to make it known. So we

¹The file described in this section has version number v2.2p and was last revised on 2005/03/29.

²Please note that if the acute accent character is active, it is necessary to take special care of coding apostrophes in a way which cannot be confounded with accents. Therefore, it is necessary to type `l'{}estri` instead of `l'estri`.

- 'e acute accented a, allowing hyphenation in the rest of the word. Valid for the following vowels: e, i, o, u (both lowercase and uppercase).
- ‘a grave accented a, allowing hyphenation in the rest of the word. Valid for the following vowels: a, e, o (both lowercase and uppercase).

Table 2: Extra definitions made by file `catalan.ldf` (activated only when using the options `activeacute` and `activegrave`)

check for the existence of `\l@catalan` to see whether we have to do something here.

```

3 \ifx\l@catalan\undefined
4 \nopatterns{Catalan}
5 \adddialect\l@catalan0
6 \fi

```

The next step consists of defining commands to switch to (and from) the Catalan language.

`\catalanhyphenmins` This macro is used to store the correct values of the hyphenation parameters `\lefthyphenmin` and `\righthyphenmin`.

```
7 \providehyphenmins{catalan}{\tw@\tw@}
```

`\captionscatalan` The macro `\captionscatalan` defines all strings used in the four standard documentclasses provided with L^AT_EX.

```

8 \addto\captionscatalan{%
9 \def\prefacename{Pr\‘oleg}%
10 \def\refname{Refer\‘encies}%
11 \def\abstractname{Resum}%
12 \def\bibname{Bibliografia}%
13 \def\chaptername{Cap\’{i}tol}%
14 \def\appendixname{Ap\‘endix}%
15 \def\contentsname{\’Index}%
16 \def\listfigurename{\’Index de figures}%
17 \def\listtablename{\’Index de taules}%
18 \def\indexname{\’Index alfab\‘etic}%
19 \def\figurename{Figura}%
20 \def\tablename{Taula}%
21 \def\partname{Part}%
22 \def\enclname{Adjunt}%
23 \def\ccname{C\‘opies a}%
24 \def\headtoname{A}%
25 \def\pagename{P\‘agina}%
26 \def\seename{Vegeu}%
27 \def\alsoname{Vegeu tamb\’e}%
28 \def\proofname{Demostraci\’o}%
29 \def\glossaryname{Glossari}%

```

30 }

`\datecatalan` The macro `\datecatalan` redefines the command `\today` to produce Catalan dates. Months are written in lowercase³.

```
31 \def\datecatalan{%
32   \def\today{\number\day~\ifcase\month\or
33     de gener\or de febrer\or de mar\c{c}\or d'abril\or de maig\or
34     de juny\or de juliol\or d'agost\or de setembre\or d'octubre\or
35     de novembre\or de desembre\fi
36     \space de~\number\year}}
```

`\extrascatalan` The macro `\extrascatalan` will perform all the extra definitions needed for the Catalan language. The macro `\noextrascatalan` is used to cancel the actions of `\extrascatalan`.

To improve hyphenation we give the grave character (') a non-zero lower case code; when we do that T_EX will find more breakpoints in words that contain this character in its rôle as apostrophe.

```
37 \addto\extrascatalan{%
38   \lccode' '= ' }
39 \addto\noextrascatalan{%
40   \lccode' '= 0 }
```

For Catalan, some characters are made active or are redefined. In particular, the " character receives a new meaning; this can also happen for the ' character and the ´ character when the options `activegrave` and/or `activeacute` are specified.

```
41 \addto\extrascatalan{\languageshorthands{catalan}}
42 \initiate@active@char{"}
43 \addto\extrascatalan{\bbl@activate{"}}
```

Because the grave character is being used in constructs such as `\catcode' '= \active` it needs to have its original category code" when the auxiliary file is being read. Note that this file is read twice, once at the beginning of the document; then there is no problem; but the second time it is read at the end of the document to check whether any labels changes. It's this second time round that the activated grave character leads to error messages.

```
44 \@ifpackagewith{babel}{activegrave}{%
45   \AtBeginDocument{%
46     \if@files\immediate\write\@auxout{\catcode096=12}\fi}
47   \initiate@active@char{'}%
48   }{}
49 \@ifpackagewith{babel}{activegrave}{%
50   \addto\extrascatalan{\bbl@activate{'}}%
51   }{}
52 \@ifpackagewith{babel}{activeacute}{%
53   \initiate@active@char{'}%
54   }{}
```

³This seems to be the common practice. See for example: E. Coromina, *El 9 Nou: Manual de redacció i estil*, Ed. Eumo, Vic, 1993

```

55 \@ifpackagewith{babel}{activeacute}{%
56 \addto\extrascatalan{\bbl@activate{'}}%
57 }{}

```

Now make sure that the characters that have been turned into shorthanfd characters expand to ‘normal’ characters outside the catalan environment.

```

58 \addto\noextrascatalan{\bbl@deactivate{}}
59 \@ifpackagewith{babel}{activegrave}{%
60 \addto\noextrascatalan{\bbl@deactivate{}}{}
61 \@ifpackagewith{babel}{activeacute}{%
62 \addto\noextrascatalan{\bbl@deactivate{'}}{}

```

Apart from the active characters some other macros get a new definition. Therefore we store the current ones to be able to restore them later. When their current meanings are saved, we can safely redefine them.

We provide new definitions for the accent macros when one or both of the options `activegrave` or `activeacute` were specified.

```

63 \addto\extrascatalan{%
64 \babel@save\"%
65 \def\"{\protect\@umlaut}}%
66 \@ifpackagewith{babel}{activegrave}{%
67 \babel@save\'%
68 \addto\extrascatalan{\def\'{\protect\@grave}}
69 }{}
70 \@ifpackagewith{babel}{activeacute}{%
71 \babel@save\'%
72 \addto\extrascatalan{\def\'{\protect\@acute}}
73 }{}

```

All the code above is necessary because we need a few extra active characters. These characters are then used as indicated in tables 1 and 2.

<code>\dieresis</code>	The original definition of <code>\"</code> is stored as <code>\dieresis</code> , because the definition of <code>\"</code> might not be the default plain T _E X one. If the user uses POSTSCRIPT fonts with the Adobe font encoding the <code>"</code> character is not in the same position as in Knuth’s font encoding. In this case <code>\"</code> will not be defined as <code>\accent"7F 1</code> , but as <code>\accent'310 #1</code> . Something similar happens when using fonts that follow the Cork encoding. For this reason we save the definition of <code>\"</code> and use that in the definition of other macros. We do likewise for <code>\'</code> , and <code>\'</code> .
<code>\textacute</code>	
<code>\textgrave</code>	

```

74 \let\dieresis\"
75 \@ifpackagewith{babel}{activegrave}{\let\textgrave\'}{}
76 \@ifpackagewith{babel}{activeacute}{\let\textacute\'}{}

```

<code>\@umlaut</code>	We check the encoding and if not using T1, we make the accents expand but
<code>\@acute</code>	enabling hyphenation beyond the accent. If this is the case, not all break positions
<code>\@grave</code>	will be found in words that contain accents, but this is a limitation in T _E X. An unsolved problem here is that the encoding can change at any time. The definitions below are made in such a way that a change between two 256-char encodings are supported, but changes between a 128-char and a 256-char encoding are not

properly supported. We check if T1 is in use. If not, we will give a warning and proceed redefining the accent macros so that T_EX at least finds the breaks that are not too close to the accent. The warning will only be printed to the log file.

```

77 \ifx\DeclareFontShape\@undefined
78 \wlog{Warning: You are using an old LaTeX}
79 \wlog{Some word breaks will not be found.}
80 \def\@umlaut#1{\allowhyphens\dieresis{#1}\allowhyphens}
81 \@ifpackagewith{babel}{activeacute}{%
82 \def\@acute#1{\allowhyphens\textacute{#1}\allowhyphens}}{}
83 \@ifpackagewith{babel}{activegrave}{%
84 \def\@grave#1{\allowhyphens\textgrave{#1}\allowhyphens}}{}
85 \else
86 \ifx\f@encoding\bbl@t@one
87 \let\@umlaut\dieresis
88 \@ifpackagewith{babel}{activeacute}{%
89 \let\@acute\textacute}{}
90 \@ifpackagewith{babel}{activegrave}{%
91 \let\@grave\textgrave}{}
92 \else
93 \wlog{Warning: You are using encoding \f@encoding\space
94 instead of T1.}
95 \wlog{Some word breaks will not be found.}
96 \def\@umlaut#1{\allowhyphens\dieresis{#1}\allowhyphens}
97 \@ifpackagewith{babel}{activeacute}{%
98 \def\@acute#1{\allowhyphens\textacute{#1}\allowhyphens}}{}
99 \@ifpackagewith{babel}{activegrave}{%
100 \def\@grave#1{\allowhyphens\textgrave{#1}\allowhyphens}}{}
101 \fi
102 \fi

```

If the user setup has extended fonts, the Ferguson macros are required to be defined. We check for their existence and, if defined, expand to whatever they are defined to. For instance, `\'a` would check for the existence of a `\@ac@a` macro. It is assumed to expand to the code of the accented letter. If it is not defined, we assume that no extended codes are available and expand to the original definition but enabling hyphenation beyond the accent. This is as best as we can do. It is better if you have extended fonts or ML- $\text{T}_{\text{E}}\text{X}$ because the hyphenation algorithm can work on the whole word. The following macros are directly derived from ML- $\text{T}_{\text{E}}\text{X}$.⁴

Now we can define our shorthands: the diaeresis and “*ela geminada*” support,

```

103 \declare@shorthand{catalan}{"i}{\textormath{\@umlaut\i}{\ddot{\imath}}}

```

⁴A problem is perceived here with these macros when used in a multilingual environment where extended hyphenation patterns are available for some but not all languages. Assume that no extended patterns exist at some site for French and that `french.sty` would adopt this scheme too. In that case, `'e` in French would produce the combined accented letter, but hyphenation around it would be suppressed. Both language options would need an independent method to know whether they have extended patterns available. The precise impact of this problem and the possible solutions are under study.

```

104 \declare@shorthand{catalan}{"l"}{\lgem{}}
105 \declare@shorthand{catalan}{"u"}{\textormath{\@umlaut u}{\ddot u}}
106 \declare@shorthand{catalan}{"I"}{\textormath{\@umlaut I}{\ddot I}}
107 \declare@shorthand{catalan}{"L"}{\Lgem{}}
108 \declare@shorthand{catalan}{"U"}{\textormath{\@umlaut U}{\ddot U}}
cedille,
109 \declare@shorthand{catalan}{"c"}{\textormath{\c c}{\prime c}}
110 \declare@shorthand{catalan}{"C"}{\textormath{\c C}{\prime C}}
'french' quote characters,
111 \declare@shorthand{catalan}{"<"}{%
112 \textormath{\guillemotleft}{\mbox{\guillemotleft}}}
113 \declare@shorthand{catalan}{">"}{%
114 \textormath{\guillemotright}{\mbox{\guillemotright}}}
grave accents,
115 \@ifpackagewith{babel}{activegrave}{%
116 \declare@shorthand{catalan}{'a'}{\textormath{\@grave a}{\grave a}}
117 \declare@shorthand{catalan}{'e'}{\textormath{\@grave e}{\grave e}}
118 \declare@shorthand{catalan}{'o'}{\textormath{\@grave o}{\grave o}}
119 \declare@shorthand{catalan}{'A'}{\textormath{\@grave A}{\grave A}}
120 \declare@shorthand{catalan}{'E'}{\textormath{\@grave E}{\grave E}}
121 \declare@shorthand{catalan}{'O'}{\textormath{\@grave O}{\grave O}}
122 \declare@shorthand{catalan}{' '}{\textquotedblleft}%
123 }{}
acute accents,
124 \@ifpackagewith{babel}{activeacute}{%
125 \declare@shorthand{catalan}{'a'}{\textormath{\@acute a}{\prime a}}
126 \declare@shorthand{catalan}{'e'}{\textormath{\@acute e}{\prime e}}
127 \declare@shorthand{catalan}{'i'}{\textormath{\@acute i}{\prime i}}
128 \declare@shorthand{catalan}{'o'}{\textormath{\@acute o}{\prime o}}
129 \declare@shorthand{catalan}{'u'}{\textormath{\@acute u}{\prime u}}
130 \declare@shorthand{catalan}{'A'}{\textormath{\@acute A}{\prime A}}
131 \declare@shorthand{catalan}{'E'}{\textormath{\@acute E}{\prime E}}
132 \declare@shorthand{catalan}{'I'}{\textormath{\@acute I}{\prime I}}
133 \declare@shorthand{catalan}{'O'}{\textormath{\@acute O}{\prime O}}
134 \declare@shorthand{catalan}{'U'}{\textormath{\@acute U}{\prime U}}
135 \declare@shorthand{catalan}{'|'}{%
136 \textormath{\csname normal@char\string\endcsname}{\prime}}
the acute accent,
137 \declare@shorthand{catalan}{' '}{%
138 \textormath{\textquotedblright}{\sp\bgroup\prim@s'}}
139 }{}
and finally, some support definitions
140 \declare@shorthand{catalan}{"-"}{\nobreak-\bbl@allowhyphens}
141 \declare@shorthand{catalan}{"|"}{%
142 \textormath{\nobreak\discretionary{-}{}{\kern.03em}%
143 \allowhyphens}{}}

```

\- All that is left now is the redefinition of \-. The new version of \- should indicate an extra hyphenation position, while allowing other hyphenation positions to be generated automatically. The standard behaviour of T_EX in this respect is unfortunate for Catalan but not as much as for Dutch or German, where long compound words are quite normal and all one needs is a means to indicate an extra hyphenation position on top of the ones that T_EX can generate from the hyphenation patterns. However, the average length of words in Catalan makes this desirable and so it is kept here.

```
144 \addto\extrascatalan{%
145 \babel@save{\-}%
146 \def\-\{\bbl@allowhyphens\discretionary{-}{-}\bbl@allowhyphens}}
```

\lgem Here we define a macro for typing the catalan “*ela geminada*” (geminated l).
 \Lgem The macros \lgem and \Lgem have been chosen for its lowercase and uppercase representation, respectively⁵.

The code used in the actual macro used is a combination of the one proposed by Feruglio and Fuster⁶ and the proposal⁷ from Valiente presented at the T_EX Users Group Annual Meeting in 1995. This last proposal has not been fully implemented due to its limitation to CM fonts.

```
147 \newdimen\leftllkern \newdimen\rightllkern \newdimen\raiselldim
148 \def\lgem{%
149 \ifmmode
150 \csname normal@char\string"\endcsname l%
151 \else
152 \leftllkern=0pt\rightllkern=0pt\raiselldim=0pt%
153 \setbox0\hbox{1}\setbox1\hbox{1/}\setbox2\hbox{.}%
154 \advance\raiselldim by \the\fontdimen5\the\font
155 \advance\raiselldim by -\ht2%
156 \leftllkern=-.25\wd0%
157 \advance\leftllkern by \wd1%
158 \advance\leftllkern by -\wd0%
159 \rightllkern=-.25\wd0%
160 \advance\rightllkern by -\wd1%
161 \advance\rightllkern by \wd0%
162 \allowhyphens\discretionary{1-}{1}%
163 {\hbox{1}\kern\leftllkern\raise\raiselldim\hbox{.}}%
164 \kern\rightllkern\hbox{1}}\allowhyphens
165 \fi
166 }
167 \def\Lgem{%
168 \ifmmode
169 \csname normal@char\string"\endcsname L%
170 \else
171 \leftllkern=0pt\rightllkern=0pt\raiselldim=0pt%
```

⁵The macro names \ll and \LL were not taken because of the fact that \ll is already used in mathematical mode.

⁶G. Valiente and R. Fuster, Typesetting Catalan Texts with T_EX, *TUGboat* **14**(3), 1993.

⁷G. Valiente, Modern Catalan Typographical Conventions, *TUGboat* **16**(3), 1995.

```

172 \setbox0\hbox{L}\setbox1\hbox{L/}\setbox2\hbox{.}%
173 \advance\raiselldim by .5\ht0%
174 \advance\raiselldim by -.5\ht2%
175 \leftllkern=-.125\wd0%
176 \advance\leftllkern by \wd1%
177 \advance\leftllkern by -\wd0%
178 \rightllkern=-\wd0%
179 \divide\rightllkern by 6%
180 \advance\rightllkern by -\wd1%
181 \advance\rightllkern by \wd0%
182 \allowhyphens\discretionary{L-}{L}%
183 {\hbox{L}\kern\leftllkern\raise\raiselldim\hbox{.}}%
184 \kern\rightllkern\hbox{L}}\allowhyphens
185 \fi
186 }

```

\l.l It seems to be the most natural way of entering the “ela geminda” to use the sequences \l.l and \L.L. These are not really macro’s by themselves but the macros \l and \L with delimited arguments. Therefor we define two macros that check if the next character is a period. If not the “polish l” will be typeset, otherwise a “ela geminada” will be typeset and the next two tokens will be ‘eaten’.

```

187 \AtBeginDocument{%
188 \let\lslash\l
189 \let\Lslash\L
190 \DeclareRobustCommand\l{\ifnextchar.\bbl@l\lslash}
191 \DeclareRobustCommand\L{\ifnextchar.\bbl@L\Lslash}}
192 \def\bbl@l#1#2{\lgem}
193 \def\bbl@L#1#2{\Lgem}

```

\up A macro for typesetting things like 1^{er} as proposed by Raymon Seroul⁸.

```

194 \DeclareRobustCommand*\up}[1]{\textsuperscript{#1}}

```

The macro \ldf@finish takes care of looking for a configuration file, setting the main language to be switched on at \begin{document} and resetting the category code of @ to its original value.

```

195 \ldf@finish{catalan}
196 \code)

```

⁸This macro has been borrowed from francais.dtx