

# Typesetting dropped capitals in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> documents

Daniel Flipo  
daniel.flipo@free.fr

## 1 Introduction

The file `lettrine.dtx`<sup>1</sup>, provides a command `\lettrine` which requires two mandatory arguments, and an optional one.

Adding `\usepackage{lettrine}` in the preamble of a document defines the command `\lettrine`, the simplest use of which is `\lettrine{<letter>}{<text>}`. It produces a dropped capital `<letter>` (2 lines high), followed by `<text>` typeset in small caps, and the rest of the paragraph is wrapped around the dropped capital. Various parameters are provided to control the size and layout of the dropped capital and match the requirements described in the books

- “Lexique des règles typographiques en usage à l’Imprimerie nationale” troisième édition (1994), ISBN-2-11-081075-0,
- “Mise en page et impression” Yves Perrousseaux, ISBN-2-911220-01-3.

The parameters can be set using David Carlisle’s `keyval.sty` syntax:

- `lines=<integer>` sets how many lines the dropped capital will occupy (default=2);
- `depth=<integer>` sets the number of lines to be reserved under the baseline, this is meant for dropped capital with positive depth, like Q (default=0);
- `lhang=<decimal>` ( $0 \leq \text{lhang} \leq 1$ ) sets how much of the dropped capital’s width should hang into the margin (default=0);
- `loversize=<decimal>` ( $-1 < \text{loversize} \leq 1$ ) enlarges the dropped capital’s height: with `loversize=0.1` its height is enlarged by 10% so that it raises above the top paragraph’s line (default=0);
- `lraise=<decimal>` does not affect the dropped capital’s height, but moves it up (if positive), down (if negative); useful with capitals like J or Q which have a positive depth, (default=0);
- `findent=<dimen>` (positive or negative) controls the horizontal gap between the dropped capital and the indented block of text (default=0pt);
- `nindent=<dimen>` shifts all indented lines, starting from the second one, horizontally by `<dimen>` (this shift is relative to the first line, default=0.5em);
- `slope=<dimen>` can be used with dropped capitals like A or V to add `<dimen>` (positive or negative) to the indentation of each line starting from the third one (no effect if `lines=2`, default=0pt);

---

<sup>1</sup>The file described in this section has version number v2.1 and was last revised on 2018/08/18.

- `ante=<text>` can be used to typeset *<text>* before the dropped capital (typical use is for French guillemets starting the paragraph).
- `image=true` (new to version 1.6) will force `\lettrine` to replace the letter normally used as dropped capital by an image in eps format (latex) or in pdf, jpg, etc. format (pdflatex); this needs the `graphicx` package to be loaded in the preamble of course. `\lettrine[image=true]{A}{n exemple}` or just `\lettrine[image]{A}{n exemple}` will load `A.eps` or `A.pdf` instead of letter A. This was suggested by Bill Jetzer. Redefining `\LettrineFont` as `\LettrineFontEPS` is no longer supported.
- `grid=true` (new to version 1.8) will force the vertical skip added above the paragraph starting with `\lettrine` to be rounded up to an integer number of `\baselineskip`. This option is meant for grid typesetting.
- `novskip=<dimen>` (new to version 1.8) overrides `\DiscardVskip` (default=0.2pt). In some cases (see options `lraise`, `loversize` or accentuated dropped capitals,...) the top of the dropped capital will raise above the top of following text (usually in small caps), this will trigger a corresponding vertical skip above the paragraph starting with `\lettrine`, *only if* this skip exceeds `\DiscardVskip`. Consider enlarging `novskip` (or `\DiscardVskip`) to prevent small vertical skips from being rounded up to `\baselineskip` when using the ‘grid’ option.
- `realheight=true` (new to version 1.9) will compute the default height of the initial so that the top of it is exactly aligned with the top of the text entered as second mandatory argument of `\lettrine` taking possible accents into account. Otherwise, the default height is computed using a customisable string `\LettrineSecondString` instead of the real argument. For backward compatibility, option `realheight` defaults to `false` and `\LettrineSecondString` to ‘x’.

You probably don’t need this option if you choose to typeset the second mandatory argument of `\lettrine` in small caps (the default). If you change `\LettrineTextFont` to `\relax` or `\upshape`, consider these two examples:

`\lettrine{H}{ello}` you probably would like the top of the ‘H’ to be aligned with the top of the ‘l’ rather than with the top of the ‘e’, adding option `realheight` does the trick: `\lettrine[realheight]{H}{ello}`.  
Global variants : `\LettrineRealHeighttrue` or (without `realheight` option) `\renewcommand*{\LettrineSecondString}{l}`.

`\lettrine{L}{a misère}` option `realheight=true` would align with the top of the ‘L’ with the top of the grave accent, the default is probably better (top of the ‘L’ aligned with the top of the non accented letters).

- `refstring`<sup>2</sup> (new to version 2.1) is meant for fancy initials with irregular heights (i.e. taken in fonts like Yinit (OpenType), `cfr-initials`,...). `refstring` forces the `\fontsize` computations to be run on the initial given as `\lettrine`’s first mandatory argument instead of the reference string `\LettrineTestString`. In most cases, this option should *not* be used: think of accentuated initials or capitals with optical correction.

---

<sup>2</sup>Unlike `grid` or `realheight`, `refstring` is not a flag (*do not* add `=true!`); it is possible to set `refstring=<string>` to override `\LettrineTestString` locally.

Example: `\lettrine[lines=4, lraise=0.1, nindent=0em, slope=-.5em]%  
{V}{oici} un exemple ...`

Coloured initials are possible in conjunction with package `color`, examples:  
`\lettrine{\textcolor{red}{A}}{n} example` or  
`\lettrine{\textcolor{gray}{0.5}{A}}{nother} one`  
see package `color` for the syntax of colour commands. Another possibility to  
colour initials globally is described below, see `\LettrineFontHook`.

Three dimensions, `\LettrineWidth`, `\LettrineHeight` and `\LettrineDepth`,  
store the final size of the initial.

Once `lettrine.sty` will be installed (run `latex` on `lettrine.ins` to extract it),  
compile and print `demo.tex` to see the possible usage of these parameters.

The default settings can be customized either in a config file `lettrine.cfg` (for a  
global usage), or on a per document basis, in the preamble of each document. The  
following list shows the syntax to set them and their default values:

- `\setcounter{DefaultLines}{2},`
- `\setcounter{DefaultDepth}{0},`
- `\renewcommand*{\DefaultLoversize}{0},`
- `\renewcommand*{\DefaultLraise}{0},`
- `\renewcommand*{\DefaultLhang}{0},`
- `\LettrineImagefalse,`
- `\LettrineOnGridfalse,`
- `\LettrineRealHeightfalse,`
- `\LettrineSelfReffalse,`
- `\LettrineVonefalse,`
- `\setlength{\DefaultFindent}{0pt},`
- `\setlength{\DefaultNindent}{0.5em},`
- `\setlength{\DefaultSlope}{0pt}.`
- `\setlength{\DiscardVskip}{0.2pt}.`

Instead of giving optional parameters to the `\lettrine` command, it is possible,  
from version 1.5, to set them on a per character basis in a second config file (sug-  
gested by Pascal Kockaert): `\renewcommand{\DefaultOptionsFile}{filename}`  
in the preamble (or anywhere in the document) will force each call to `\lettrine`  
to read the file *filename*. See examples of such config files in the subdirectory  
`contrib`.

The dimensional parameters `findent`, `nindent` and `slope`, can be set in *filename*  
relative to `\LettrineWidth` if needed. The settings read from this file will be  
overridden by the optional arguments eventually given to `\lettrine`.

`\LettrineTextFont` sets the font used for the second argument of `\lettrine`, its default definition is `\newcommand{\LettrineTextFont}{\scshape}` (second argument in small caps, this can be changed using `\renewcommand`).

`\LettrineFont` sets the font used for the dropped capital, usually the current font in a (large) size, computed automatically from the number of lines it will fill: the font size is computed so that, a *standard* dropped capital (say Z, not Å) when sitting on its baseline, gets its top aligned with the top of the following text (provided `loversize = 0` and `lines ≥ 2`). When `lines=1`, size is computed as if `lines` was 2.

A hook `\LettrineFontHook` is provided to change the font used for the dropped capital, syntax follows L<sup>A</sup>T<sub>E</sub>X's low-level font interface (see L<sup>A</sup>T<sub>E</sub>X Companion, p.187–192), the `\selectfont` command is issued by `\LettrineFont`:

```
\renewcommand{\LettrineFontHook}{\fontfamily{ppl}\fontseries{bx}}%
                                \fontshape{sl}},
```

selects Palatino bold expanded slanted for the dropped capital.

`\LettrineFontHook` can also be used to change the colour of all lettrines in a (part of) document: `\renewcommand{\LettrineFontHook}{\color[gray]{0.5}}`

will colour the initials following this command in grey.

**Important notice:** the sizing works fine with *fully scalable* fonts (like the standard PostScript fonts), but might not work well with CM/EC fonts which have two limitations: only a limited number of sizes is available by default (precise adjustments are impossible), and the largest size (25pt or 35pt) is often too small. The CM fonts are now available in PostScript type1 format for free (courtesy of BlueSky/Y&Y), to make them fully scalable, it is mandatory to add `\usepackage{type1cm}` in the preamble of your document. The EC fonts are also available in type1 format for free (thanks to Vladimir Volovich, they are called cm-super), and adding `\usepackage{type1ec}`<sup>3</sup> in the preamble will make them fully scalable too. So, if you want `lettrine.sty` to work properly with CM or EC fonts, you will need *PostScript versions* of these fonts *and* one of the packages `type1cm.sty` or `type1ec.sty`.

The LM fonts are a good replacement for both CM and EC fonts they are fully scalable, so you should use them instead of CM or EC fonts. `\usepackage{lmodern}` is the command to switch them on (add `\usepackage[T1]{fontenc}` when composing in one of the western languages other than English in order to get proper hyphenation).

You can also consider using one of the standard PostScript fonts (Palatino, Times, Utopia...), or any OpenType font, they are fully scalable too!

#### Known issues:

- nothing is done to prevent page-breaking in a paragraph starting with a dropped capital; when it happens to hang into the footer, page-breaking has to be done manually;
- `\lettrine` works within ‘quote’ ‘quotation’, ‘abstract’ environments but does not work within ‘center’ environments (except with option `[lines=1]`);

---

<sup>3</sup>This package, available on CTAN, was first released on 2002/07/30.

- `\lettrine` does not work within lists;
- if a *list* has to be included in a paragraph starting with `\lettrine`, it is necessary to add the command `\parshape=0` just after the end of the list (starting a new paragraph just before or just after the list works too). Remember that ‘quote’, ‘quotation’, ‘abstract’ environments are implemented as *lists* in L<sup>A</sup>T<sub>E</sub>X.
- if you are facing some slight height inaccuracy for a dropped capital, you can try option `refstring`; this option is meant for fancy (unaccented) initials. Informations about targeted and effective initial’s height are available in the `.log` file. Using LuaTeX or XeTeX engines with OpenType fonts may be an option (some TFM files for Type1 fonts are slightly inaccurate).
- `\LettrineTestString`’s value has changed over the time; to achieve strict backward compatibility with previous versions, please look at the end of `lettrine.cfg` and consider adding the `Vone` option to the `\lettrine` command or setting `\LettrineVone>true` for global action on the whole document.

## 2 T<sub>E</sub>Xnical details

This package only runs with L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> and requires `keyval.sty`

```
1 \NeedsTeXFormat{LaTeX2e}[1999/12/01]
2 \RequirePackage{keyval,minifp}
```

Default initializations: define the necessary counters, lengths, and commands to hold the default settings and set these default settings. They can be overwritten in file `lettrine.cfg`.

```
3 \newcounter{DefaultLines}
4 \setcounter{DefaultLines}{2}
5 \newcounter{DefaultDepth}
6 \newcommand*{\DefaultOptionsFile}{\relax}
7 \newcommand*{\DefaultLoversize}{0}
8 \newcommand*{\DefaultLraise}{0}
9 \newcommand*{\DefaultLhang}{0}
10 \newdimen\DefaultFindent
11 \setlength{\DefaultFindent}{\z@}
12 \newdimen\DefaultNindent
13 \setlength{\DefaultNindent}{0.5em}
14 \newdimen\DefaultSlope
15 \setlength{\DefaultSlope}{\z@}
16 \newdimen\DiscardVskip
17 \setlength{\DiscardVskip}{0.2\p@}
18 \newif\ifLettrineImage
19 \newif\ifLettrineOnGrid
20 \newif\ifLettrineRealHeight
21 \newif\ifLettrineVone
22 \newif\ifLettrineSelfRef
```

Then let’s define the necessary internal counters, lengths, and commands.

```

23 \newsavebox{\L@lbox}
24 \newsavebox{\L@tbox}
25 \newcounter{L@lines}
26 \newcounter{L@depth}
27 \newdimen\L@Pindent
28 \newdimen\L@Findent
29 \newdimen\L@Nindent
30 \newdimen\L@lraise
31 \newdimen\L@first
32 \newdimen\L@next
33 \newdimen\L@slope
34 \newdimen\L@height
35 \newdimen\L@novskip
36 \newdimen\L@target@ht
37 \newdimen\L@target@dp
38 \newdimen\L@target@tht
39 \newdimen\LettrineWidth
40 \newdimen\LettrineHeight
41 \newdimen\LettrineDepth
42 \newcommand*{\L@file}{}
43 \newcommand*{\L@hang}{}
44 \newcommand*{\L@oversize}{}
45 \newcommand*{\L@raise}{}
46 \newcommand*{\L@ante}{}
47 \newif\ifL@image
48 \newif\ifL@grid
49 \newif\ifL@realh
50 \newif\ifL@vone
51 \newif\ifL@selfref

```

Provide commands for the fonts used to typeset the two mandatory arguments of `\lettrine`.

`\LettrineTextFont` In French, small caps usually follow the dropped capital.

```

52 \newcommand*{\LettrineTextFont}{\scshape}

```

`\LettrineTestString` The default size for the dropped capital is computed so that the top of it is exactly  
`\LettrineSecondString` aligned with the top of the following text; an extra height (positive or negative)  
`\LettrineFontHook` may be added globally by redefining `\Defaultloversize` or locally using optional  
`\LettrineFont` argument `loversize=`. If `lines=1`, the default size for the dropped capital is  
computed as if `lines=2`.

`\Lettrine@height` computes the targeted height for the dropped capital and stores it into `\L@target@tht` which depends only on `L@lines` and on the height of `\L@tbox` (see fig. 1). So options *must* be read and `\L@tbox` must be properly initialised *before* executing `\Lettrine@height` (see below in `\@lettrine` code).

`\L@height` is set to `\L@target@tht` raised by the `\L@oversize` factor.

```

53 \newcommand*{\LettrineTestString}{EFTZ}
54 \newcommand*{\LettrineSecondString}{x}
55 \def\Lettrine@height{%
56   \setlength{\L@target@ht}{\ht\L@tbox}%

```

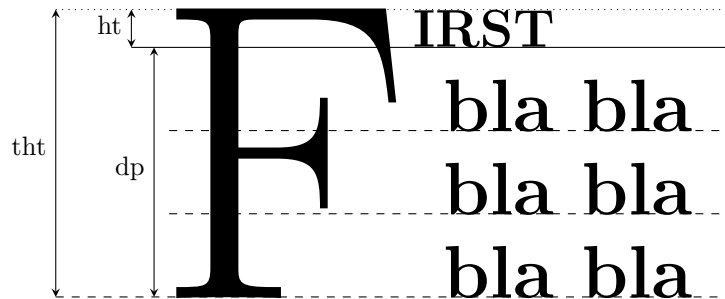


Figure 1: Initial's targeted dimensions ( $\backslash\text{lines} = 4$ )

As  $\backslash\text{baselineskip}$  might be a rubber length, let's convert it into a 'dimen' using  $\backslash\text{@tempdima}$ .

```

57 \@tempdima=\baselineskip
58 \setlength{\L@target@dp}{\value{L@lines}\@tempdima}%
59 \ifnum\value{L@lines}>1
60   \addtolength{\L@target@dp}{-\@tempdima}%
61 \else
62   \addtolength{\L@target@ht}{\L@target@dp}%
63   \setlength{\L@target@dp}{0pt}%
64 \fi
65 \setlength{\L@target@tht}{\L@target@ht}%
66 \addtolength{\L@target@tht}{\L@target@dp}%
67 \setlength{\L@height}{\L@target@tht}%
68 \addtolength{\L@height}{\L@oversize\L@target@tht}%
69 \ifL@image
70 \else

```

$\backslash\text{L@height}$  now holds the exact height required for the dropped capital, let's compute the matching  $\backslash\text{fontsize}$ 's value (unless the initial is a picture); we first measure the height of a "standard" capital (i.e. either listed in  $\backslash\text{LettrineTestString}$  or the initial itself) scaled by  $\backslash\text{fontsize}$  with argument  $\backslash\text{L@height}$ . Mind that some font designers apply optical correction to capitals C, G, O, or Q (they are slightly taller than 'T' or 'Z'), that's why they are not part of  $\backslash\text{LettrineTestString}$ . When optional argument **selfref** is **true** the initial itself is taken as reference to compute  $\backslash\text{fontsize}$ , this can be handy when working with fancy fonts (i.e. cfr-initials, Yinit).

```

71 \ifL@selfref
72   \def\Lettrine@RefString{\l@refstring}%
73 \else
74   \def\Lettrine@RefString{\LettrineTestString}%
75 \fi
76 \sbox{\@tempboxa}{\LettrineFontHook
77   \fontsize{\L@height}{\L@height}\selectfont
78   \Lettrine@RefString}%
79 \@tempcnta=ht\@tempboxa
80 \@tempcntb=\L@height

```

$\backslash\text{@tempcntb}$  holds the target height converted into sp (integer) and  $\backslash\text{@tempcnta}$

holds the height in sp of the “standard” initial scaled by `\fontsize` with `L@height` as argument; let’s divide `@tempcntb` by `@tempcnta` to compute a (4 decimal accurate) scaling factor `L@factor`. `\fontsize`’s argument providing the requested `L@height` is `L@factorL@height`.

Up to v2.01, possible values for `L@factor` were either 1 or any value  $\geq 1.1$  which was fine for almost every font but not all (i.e. `cfr-initials`). Starting with v2.1, `L@factor` is computed more accurately by `minifp.sty`; option `Vone` is provided for backward compatibility.

```

81      \ifL@vone
82      \multiply\@tempcntb by 100
83      \divide\@tempcnta by 100
84      \divide\@tempcntb by \@tempcnta
85      \advance\@tempcntb by -9999
86      \ifnum\@tempcntb>0
87      \def\L@factor{1.\the\@tempcntb}%
88      \else
89      \def\L@factor{1}%
90      \fi
91      \else
92      \MFPdiv{\the\@tempcntb}{\the\@tempcnta}\L@factor
93      \fi
94      \fi
95 }

```

`\LettrineFontHook` enables to select another font for the dropped capital. Its default definition is empty (the current text font is used).

```

96 \newcommand*\LettrineFontHook{}
97 \newcommand*\LettrineFont{%
98   \LettrineFontHook
99   \fontsize{L@factorL@height}{L@factorL@height}%
100  \selectfont
101 }

```

`\LettrineFontEPS` The following definition is for use with dropped capitals defined as images: EPS, PDF, JPG, PNG files (see examples in `demo.tex`). Its use requires the `graphicx` package to be loaded in the preamble with `\usepackage{graphicx}`. The required size is computed just as in the standard case, `\includegraphics` prints the EPS file at this size.

```

102 \newcommand*\LettrineFontEPS{%
103   \LettrineFontHook\includegraphics[height=L@height]%
104 }

```

Set up keyval initializations.

```

105 \define@key{L}{lines}{\setcounter{L@lines}{#1}}
106 \define@key{L}{depth}{\setcounter{L@depth}{#1}}
107 \define@key{L}{lhang}{\renewcommand*\L@hang{#1}}
108 \define@key{L}{loversize}{\renewcommand*\L@oversize{#1}}
109 \define@key{L}{lraise}{\renewcommand*\L@raise{#1}}
110 \define@key{L}{ante}{\renewcommand*\L@ante{#1}}

```



```

111 \define@key{L}{findent}{\setlength{\L@Findent}{#1}}
112 \define@key{L}{nindent}{\setlength{\L@Nindent}{#1}}
113 \define@key{L}{slope}{\setlength{\L@slope}{#1}}
114 \define@key{L}{image}[true]{\csname L@image#1\endcsname}
115 \define@key{L}{grid}[true]{\csname L@grid#1\endcsname}
116 \define@key{L}{realheight}[true]{\csname L@realh#1\endcsname}
117 \define@key{L}{novskip}{\setlength{\L@novskip}{#1}}
118 \define@key{L}{refstring}[\l@initial]{\L@selfreftrue
119                                     \def\l@refstring{#1}}
120 \define@key{L}{Vone}[true]{\csname L@vone#1\endcsname}

```

`\LettrineOptionsFor` This command is to be used in an optional config file (the name of which is found in `\DefaultOptionsFile`) to set the values of parameters on a per character basis, for instance:

`\LettrineOptionsFor{A}{slope=0.6em, findent=-1em, nindent=0.6em}` creates an internal command (`\l@A-keys` in this example), which expands to the options given as second argument of `\LettrineOptionsFor` for letter ‘A’.

```

121 \newcommand*{\LettrineOptionsFor}[2]{\@namedef{l@#1-keys}{#2}}

```

`\lettrine` Now let’s define `\lettrine`.

```

122 \def\lettrine{\@ifnextchar[\@lettrine{\@lettrine[]}}
123 \def\@lettrine[#1]#2#3{%
124   \def\l@initial{#2}\let\l@refstring\l@initial

```

First reset the parameters to their default values:

```

125   \setcounter{L@lines}{\value{DefaultLines}}%
126   \setcounter{L@depth}{\value{DefaultDepth}}%
127   \renewcommand*{\L@hang}{\DefaultLhang}%
128   \renewcommand*{\L@oversize}{\DefaultLoversize}%
129   \renewcommand*{\L@raise}{\DefaultLraise}%
130   \renewcommand*{\L@ante}{}%
131   \setlength{\L@Findent}{\DefaultFindent}%
132   \setlength{\L@Nindent}{\DefaultNindent}%
133   \setlength{\L@slope}{\DefaultSlope}%
134   \setlength{\L@novskip}{\DiscardVskip}%
135   \ifLettrineImage\L@imagetrue\else\L@imagefalse\fi
136   \ifLettrineOnGrid\L@gridtrue\else\L@gridfalse\fi
137   \ifLettrineRealHeight\L@realhtrue\else\L@realhfalse\fi
138   \ifLettrineVone\L@vonetrue\else\L@vonefalse\fi
139   \ifLettrineSelfRef\L@selfreftrue\else\L@selfreffalse\fi

```

`\LettrineFont` and `\LettrineFontEPS` both rely on `\L@height` which depends on `\L@tbox`, thus requiring prior execution of `\Lettrine@height`. The content of `\L@tbox` depends on option `realheight`, so we have to read `\lettrine`’s optional argument and initialise the `\L@tbox` content now <sup>4</sup>.

```

140   \setkeys{L}{#1}%
141   \sbox{\L@tbox}{\LettrineTextFont{\LettrineSecondString}}%
142   \ifL@realh
143     \def\@tempa{#3}

```

---

<sup>4</sup>Now means before eventually reading the config file.

```

144 \ifx\@tempa\@empty
145 \PackageWarning{lettrine.sty}%
146 {Empty second argument,\MessageBreak
147 ignoring option `realheight';}%
148 \else
149 \sbox{\L@tbox}{\LettrineTextFont{#3}}%
150 \fi
151 \fi

```

Then try to read an optional file (its name is given by `\DefaultOptionsFile`), do this inside a group, and define a global command `\l@LOKeys` which will expand to the list of options given by `\LettrineOptionsFor` for the current initial (defined by #2)...

```

152 \if\DefaultOptionsFile\relax
153 \else
154 \begingroup
155 \InputIfFileExists{\DefaultOptionsFile}%
156 {}%
157 {\PackageWarning{lettrine.sty}%
158 {File \DefaultOptionsFile\space not found}%
159 }%

```

Gobble the colour commands, just keep the letter argument.

```

160 \def\color##1##{\l@color{##1}}%
161 \let\l@color\@gobbletwo
162 \def\textcolor##1##{\l@textcolor{##1}}%
163 \def\l@textcolor##1##2##3{##3}%

```

Save the list of options relevant to the letter in #2 in `\l@LOKeys` (list is empty eventually).

```

164 \expandafter\ifx\csname l@#2-keys\endcsname\relax
165 \gdef\l@LOKeys{}%
166 \else
167 \xdef\l@LOKeys{\csname l@#2-keys\endcsname}%
168 \fi
169 \endgroup

```

Now apply these options (the following code executes `\setkeys{L}{\l@LOKeys}` where `\l@LOKeys` is expanded, see `keyval.sty`).

```

170 \def\KV@prefix{KV@L@}%
171 \let\@tempc\relax
172 \expandafter\KV@do\l@LOKeys,\relax,

```

As some parameters' values `findent`, `nindent` and `slope` — which do not influence the initial's size — may be given relative to `\LettrineWidth`, this has to be done again after measuring the lettrine for `\LettrineWidth` to be set properly.

```

173 \Lettrine@height
174 \sbox{\L@lbox}{\LettrineFont #2}%
175 \setlength{\LettrineWidth}{\wd\L@lbox}%
176 \def\KV@prefix{KV@L@}%
177 \let\@tempc\relax
178 \expandafter\KV@do\l@LOKeys,\relax,

```

As local options prevail on those held in the config file, we have to read local options again:

```
179   \setkeys{L}{#1}%
180 \fi
```

Options and optional config file have been taken into account, we can now save the first mandatory argument of `\lettrine` properly scaled into `\L@lbox`. Depending on the boolean `image`, `\LettrineFont` or `\LettrineFontEPS` is used.

```
181 \Lettrine@height
182 \ifL@image
183   \sbox{\L@lbox}{\LettrineFontEPS{#2}}%
184 \else
185   \sbox{\L@lbox}{\LettrineFont {#2}}%
186 \fi
```

Store the initial's final dimensions,

```
187 \setlength{\LettrineWidth}{\wd\L@lbox}%
188 \setlength{\LettrineHeight}{\ht\L@lbox}%
189 \setlength{\LettrineDepth}{\dp\L@lbox}%
```

print some informations about accuracy to the log file,

```
190 \begingroup
191 \def\IeC##1{##1}%
192 \@tempdima=\L@oversize pt\relax
193 \PackageInfo{lettrine.sty}%
194   {Targeted height = \the\L@target@tth\MessageBreak
195     (no accent, loversize=0),\MessageBreak
196     Lettrine height = \the\LettrineHeight\space (#2)%
197     \ifdim\@tempdima=\z@ \else \space loversize=\L@oversize\fi;%
198     \MessageBreak reported}%
199 \endgroup
```

and reset `\L@tbox`'s content (mandatory in case `realheight=false`):

```
200 \sbox{\L@tbox}{\LettrineTextFont{#3}}%
```

Start a new paragraph, skipping the necessary amount of space if the dropped capital sticks out of the top of paragraph. We use `\L@first` to compute the amount of space to be skipped.

The basis for `\L@raise` (and `\L@oversize`, see `\LettrineFont`) is `\L@target@tth`.

```
201 \setlength{\L@first}{\LettrineHeight}%
202 \setlength{\L@lraise}{\L@raise\L@target@tth}%
203 \addtolength{\L@first}{\L@lraise}%
204 \ifnum\value{L@lines}=1
205   \addtolength{\L@first}{-\ht\L@tbox}%
206 \else
207   \addtolength{\L@first}{-\L@target@tth}%
208   \addtolength{\L@lraise}{-\L@target@dp}%
209 \fi
210 \par
```

\L@first now holds the height of the needed \vskip; if too small it will be discarded.

```
211 \ifdim\L@first>\L@novskip
```

When the `grid` option is `true`, let's round up \L@first to the next integer number of \baselineskip.

```
212 \ifL@grid
213 \tempdima=\baselineskip
214 \tempdimb=\tempdima
215 \advance\tempdimb by \L@novskip
216 \tempcnta=1
217 \loop\ifdim\L@first>\tempdimb
218 \advance\tempcnta by 1
219 \advance\L@first by -\tempdima
220 \repeat
221 \L@first=\tempcnta\baselineskip
222 \fi
223 \vskip\L@first
224 \fi
```

Again, we (mis)use the length \L@first to compute the width of the text eventually coming before the dropped capital. It is reset later on to hold the first line's length.

```
225 \setlength{\L@Pindent}{\wd\L@lbox}%
226 \addtolength{\L@Pindent}{-\L@hang\wd\L@lbox}%
227 \settowidth{\L@first}{\L@ante}%
228 \addtolength{\L@Pindent}{\L@first}%
229 \addtolength{\L@Pindent}{\L@Findent}%
230 \setlength{\L@first}{\linewidth}%
231 \addtolength{\L@first}{-\L@Pindent}%

```

Now let's compute \L@Nindent and \L@next for the next lines.

```
232 \addtolength{\L@Nindent}{\L@Pindent}%
233 \setlength{\L@next}{\linewidth}%
234 \addtolength{\L@next}{-\L@Nindent}%

```

This is for quotation, quote, abstract... environments: \linewidth is set by these environments, all we have to do is to shift our text left by \rightmargin (amount of space locally added to \leftmargin in these environments).

```
235 \addtolength{\L@Pindent}{\rightmargin}%
236 \addtolength{\L@Nindent}{\rightmargin}%

```

Now, set up the shape of the new paragraph (designed by \parshape).

```
237 \addtocounter{L@lines}{1}%
238 \addtocounter{L@lines}{\value{L@depth}}%
239 \def\L@parshape{\c@L@lines \the\L@Pindent \the\L@first}%
240 \@tempcnta=\tw@
241 \@whilenum \@tempcnta<\c@L@lines\do{%
242 \edef\L@parshape{\L@parshape \the\L@Nindent \the\L@next}%
243 \addtolength{\L@Nindent}{\L@slope}%
244 \addtolength{\L@next}{-\L@slope}%
245 \advance\@tempcnta\@ne}%
246 \edef\L@parshape{\L@parshape \rightmargin \the\linewidth}%

```

```

247 \noindent
248 \parshape=\L@parshape\relax

```

Write the dropped capital into the left margin, and wrap the rest of paragraph around it.

```

249 \smash{\llap{\mbox{\L@ante}\raisebox{\L@lraise}{\usebox{\L@lbox}}}%
250         \hskip \the\L@Findent}}%
251 \usebox{\L@tbox}%
252 }

```

This ends the definition of `\lettrine`.

Load a local config file if present in L<sup>A</sup>T<sub>E</sub>X's search path.

```

253 \InputIfFileExists{lettrine.cfg}
254   {\typeout{Loading lettrine.cfg}}
255   {\typeout{lettrine.cfg not found, using default values}}

```

### 3 Configuration file

```
256 %% lettrine.cfg: configuration file for lettrine.sty
257 %%
258 %% If you want to customize lettrine, please *do not* hack into the
259 %% code, copy this file to your working directory and customize the
260 %% copy as you like.
261 %%
262 %% Uncomment any of these lines and change the parameters' values
263 %% to fit your needs (see lettrine.dtx).
264 %%
265 %%\setcounter{DefaultLines}{2}
266 %%\setcounter{DefaultDepth}{0}
267 %%
268 %% These are *decimal* numbers:
269 %%\renewcommand*{\DefaultLoversize}{0}
270 %%\renewcommand*{\DefaultLraise}{0}
271 %%\renewcommand*{\DefaultLhang}{0}
272 %%
273 %% These are *lengths* (don't forget the unit):
274 %%\setlength{\DefaultFindent}{0pt}
275 %%\setlength{\DefaultNindent}{0.5em}
276 %%\setlength{\DefaultSlope}{0mm}
277 %%\setlength{\DiscardVskip}{0.2pt}
278 %%
279 %% These are *flags* (value=true/false):
280 %%\LettrineImagefalse
281 %%\LettrineOnGridfalse
282 %%\LettrineRealHeightfalse
283 %%\LettrineSelfReffalse
284 %%\LettrineVonefalse
285 %%
286 %% These are *commands* (value=string, only height matters):
287 %%\renewcommand*{\LettrineTestString}{EFTZ}
288 %%\renewcommand*{\LettrineSecondString}{x}
289 %% For backward compatibility with versions 1.63 to 1.9:
290 %%\renewcommand*{\LettrineTestString}{ABCDEFGHJKLMNOPQRSTUVWXYZ}
291 %% For backward compatibility with versions up to 1.62:
292 %%\renewcommand*{\LettrineTestString}{X}
293 %%
294 %% In case you want to set parameters for some letters
295 %% in file `optfile.cfl'
296 %% \renewcommand{\DefaultOptionsFile}{optfile.cfl}
```

## 4 Change History

Changes are listed in reverse order (latest first) from version 1.0

<b>v2.1</b>	General: Newif \ifLettrineSelfRef and new option ‘refstring’. . . . . 5 Newif \ifLettrineVone and new option ‘Vone’. . . . . 5 \LettrineFont: \Lettrine@height moved out of \LettrineFont: \global settings no longer required. \fontsize computations moved into \Lettrine@height. . . . . 6 Computation of \L@factor for \fontsize done by the minifp package. . . . . 8	\value{L@depth}. . . . . 12 Use the second mandatory argument of \lettrine or \LettrineSecondString (which defaults to ‘x’) to compute \L@height. This is controlled by the ‘realheight’ flag. . . . . 9 \LettrineFont: \theL@lines changed to \value{L@lines}. Needed for babel-hebrew which redefines \@arabic. . . . . 6
<b>v2.0</b>	\lettrine: Add informations about targeted and actual height of the initial to the .log file. . . . . 11 \LettrineFont: \LettrineTestString changed from ‘ABCDEFGHJKLMNO- QRSTUVWXYZ’ to ‘EFTZ’ as some capitals like C, G, O, Q or X might be slightly taller (possible optical correction). . . . 6 Change factors 100/100 to 50/200 to reach larger sizes (initial’s height up to 20cm). . . . 8 Store targeted dimensions of the dropped capital (ht, dp, tht) for further use. . . . . 6	<b>v1.8</b> General: Added newif \ifLettrineOnGrid and new dimen \DiscardVskip, default (0.2pt) set for compatibility with previous releases. . . . . 5 Added two keyval options: ‘grid’ (true/false) and ‘novskip’ to override \DiscardVskip. . . . . 2 \lettrine: The 0.2pt limit for discarded vskips is now customisable through \DiscardVskip and option ‘novskip’. . . . . 11
<b>v1.9</b>	General: New customisable string \LettrineSecondString to tune the initial’s height. . . . . 3 New keyval option: ‘realheight’ (true/false) and new global flag \ifLettrineRealHeight. . . . . 2 \lettrine: \theDefaultLines changed to \value{DefaultLines}, same with \theDefaultDepth. Needed for babel-hebrew which redefines \@arabic. Thanks to Ulrike Fischer for providing the fix. . . . . 9 \theL@depth changed to	<b>v1.7</b> General: New counter to add lines for dropped capitals with positive depth, like Q. . . . . 1 <b>v1.65</b> \lettrine: Measure and store the initial’s final dimensions. . . . . 11 <b>v1.64</b> \lettrine: Remove \$ around \smash and add \relax. Bug pointed out by David Monniaux. Correction by Enrico Gregorio. 13 <b>v1.63</b> \LettrineFont: Added command \LettrineTestString which defaults to ‘ABCDEFGHI- JKLMNOQRSTUVWXYZ’. In previous versions height computations were based on letter ‘X’ which might not exist in some (rare) fonts. Pointed out by Raphaël Pinson. . . . . 6

<b>v1.6</b>	General: Add a flag to switch to images in eps or pdf format. Suggested by Bill Jetzer. . . . .	2			
	Added newif \ifL@grid. . . . .	5	<b>v1.3</b>		
	Added newif \ifL@image. . . . .	5		General: Correct the documentation to mention the cm-super fonts and the typelec package by Vladimir Volovich. . . . .	4
	Added newif \ifLettrineImage. . . . .	5	<b>v1.2</b>		
	\lettrine: Add braces around #3 to allow commands taking an argument (such as \MakeLowercase) in \LettrineTextFont. Suggested by Philipp Lehman. . . . .	11		General: \newlength changed to \newdimen, to correct a bug with seminar.cls (pointed out by Peter Münster). . . . .	5
	\LettrineFontEPS: Added \LettrineFontHook to \LettrineFontEPS. This is needed for color options. . . . .	8		\lettrine: \baselineskip may be a rubber length, we convert it to a dimen. . . . .	11
<b>v1.5</b>	General: \LettrineOptionsFor and \LettrineWidth added. . . . .	9		\LettrineFont: \baselineskip may be a rubber length, we convert it to a dimen. . . . .	6
	\lettrine: Added reading of an optional config file \DefaultOptionsFile. . . . .	10	<b>v1.1</b>		
<b>v1.4</b>	\lettrine: \lettrine still didn't work properly in quote, quotation, abstract environments, pointed out by Matthias C. Schmidt.			\lettrine: Add \rightmargin to \L@Pindent for \Lettrine to work properly in quote, quotation, abstract environments... but do not change \linewidth which is set by these environments. . . . .	12
				\rightmargin was added too early to \L@Nindent, thus making \L@next too short by \rightmargin. . . . .	12