

Typesetting dropped capitals in L^AT_EX 2_ε documents

Daniel Flipo
daniel.flipo@free.fr

1 Introduction

The file `lettrine.dtx`¹, 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);

¹The file described in this section has version number v2.0 and was last revised on 2018/07/21.

- **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` still works for compatibility but is deprecated.
- **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 ‘ll’ 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).

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} exemple` 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,`
- `\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 (suggested 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 \LaTeX 's low-level font interface (see \LaTeX 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}`² 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]`);
- `\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^AT_EX.
- if you are facing some slight height inaccuracy for a dropped capital, you can try changing `\LettrineTestString`:
`\renewcommand*{\LettrineTestString}{<letter>}`

²This package, available on CTAN, was first released on 2002/07/30.

where $\langle letter \rangle$ is the used dropped capital *without accent*. 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`.

2 T_EXnical details

This package only runs with L^AT_EX 2_ε and requires `keyval.sty`

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

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
```

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

```
21 \newsavebox{\L@lbox}
22 \newsavebox{\L@tbox}
23 \newcounter{L@lines}
24 \newcounter{L@depth}
25 \newdimen\L@Pindent
26 \newdimen\L@Findent
27 \newdimen\L@Nindent
28 \newdimen\L@lraise
29 \newdimen\L@first
30 \newdimen\L@next
31 \newdimen\L@slope
32 \newdimen\L@height
```

```

33 \newdimen\L@novskip
34 \newdimen\L@target@ht
35 \newdimen\L@target@dp
36 \newdimen\L@target@tht
37 \newdimen\LettrineWidth
38 \newdimen\LettrineHeight
39 \newdimen\LettrineDepth
40 \newcommand*{\L@file}{}
41 \newcommand*{\L@hang}{}
42 \newcommand*{\L@oversize}{}
43 \newcommand*{\L@raise}{}
44 \newcommand*{\L@ante}{}
45 \newif\ifL@image
46 \newif\ifL@grid
47 \newif\ifL@realh

```

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

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

```

48 \newcommand*{\LettrineTextFont}{\scshape}

```

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

`\LettrineFont`

`\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`. So options *must* be read and `\L@tbox` must be properly initialised *before* executing `\Lettrine@height` (see below in `\@lettrine` code). A default initialisation of `\L@tbox` is provided just in case `\LettrineFont` would be used outside `\lettrine`.

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

```

49 \def\Lettrine@height{%
50   \ifvoid\L@tbox
51     \sbox{\L@tbox}{\LettrineTextFont{\LettrineSecondString}}%
52   \fi
53   \global\setlength{\L@target@ht}{\ht\L@tbox}%

```

As `\baselineskip` might be a rubber length, let's convert it into a 'dimen' using `\@tempdima`.

```

54   \@tempdima=\baselineskip
55   \global\setlength{\L@target@dp}{\value{L@lines}\@tempdima}%
56   \ifnum\value{L@lines}>1
57     \global\addtolength{\L@target@dp}{-\@tempdima}%
58   \else
59     \global\addtolength{\L@target@ht}{\L@target@dp}%
60     \global\setlength{\L@target@dp}{0pt}%
61   \fi
62   \global\setlength{\L@target@tht}{\L@target@ht}%

```

```

63 \global\addtolength{\L@target@tth}{\L@target@dp}%
64 \setlength{\L@height}{\L@target@tth}%
65 \addtolength{\L@height}{\L@oversize\L@target@tth}%
66 }

```

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

```

67 \newcommand*{\LettrineFontHook}{}
68 \newcommand*{\LettrineTestString}{EFTZ}
69 \newcommand*{\LettrineSecondString}{x}
70 \newcommand*{\LettrineFont}{%
71   \Lettrine@height

```

`\L@height` now holds the exact height required for the dropped capital, let's compute the matching `\fontsize`'s value: we divide `\L@height` (converted into sp) by the “standard” capitals' height (see the list in `\LettrineTestString`) to get a scaling factor (always ≥ 1). All capitals in `\LettrineTestString` are expected to have the same height, in case this assumption would be wrong for some special font, `\LettrineTestString` can be customised to any non empty subset of capitals, the maximal height of these capitals will be taken into account. Mind that some fonts 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 `\LettrineTestString`.

```

72   \sbox{\@tempboxa}{\LettrineFontHook\fontsize{\L@height}{\L@height}%
73   \selectfont \LettrineTestString}%

```

Arithmetic calculations convert the dimensions into integers (in sp) and compute a (4 decimal accurate) scaling factor.

```

74   \@tempcntb=\ht\@tempboxa
75   \@tempcnta=\L@height
76   \multiply\@tempcnta by 50
77   \divide\@tempcntb by 200
78   \divide\@tempcnta by \@tempcntb
79   \advance\@tempcnta by -9999
80   \ifnum\@tempcnta>0
81     \def\@tempa{1.\the\@tempcnta}%
82   \else
83     \def\@tempa{1}%
84   \fi
85   \LettrineFontHook
86   \fontsize{\@tempa\L@height}{\@tempa\L@height}%
87   \selectfont
88 }

```

`\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.

```

89 \newcommand*{\LettrineFontEPS}{%
90   \Lettrine@height\LettrineFontHook

```

```

91 \includegraphics[height=\L@height]%
92 }

```

Set up keyval initializations.

```

93 \define@key{L}{lines}{\setcounter{L@lines}{#1}}
94 \define@key{L}{depth}{\setcounter{L@depth}{#1}}
95 \define@key{L}{lhang}{\renewcommand*{\L@hang}{#1}}
96 \define@key{L}{loversize}{\renewcommand*{\L@oversize}{#1}}
97 \define@key{L}{lraise}{\renewcommand*{\L@raise}{#1}}
98 \define@key{L}{ante}{\renewcommand*{\L@ante}{#1}}
99 \define@key{L}{findent}{\setlength{\L@Findent}{#1}}
100 \define@key{L}{nindent}{\setlength{\L@Nindent}{#1}}
101 \define@key{L}{slope}{\setlength{\L@slope}{#1}}
102 \define@key{L}{image}[true]{\csname L@image#1\endcsname}
103 \define@key{L}{grid}[true]{\csname L@grid#1\endcsname}
104 \define@key{L}{realheight}[true]{\csname L@realh#1\endcsname}
105 \define@key{L}{novskip}{\setlength{\L@novskip}{#1}}

```

`\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’.

```

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

```

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

```

107 \def\lettrine{\@ifnextchar[\@lettrine{\@lettrine[]}}
108 \def\@lettrine[#1]#2#3{%

```

First reset the parameters to their default values:

```

109 \setcounter{L@lines}{\value{DefaultLines}}%
110 \setcounter{L@depth}{\value{DefaultDepth}}%
111 \renewcommand*{\L@hang}{\DefaultLhang}%
112 \renewcommand*{\L@oversize}{\DefaultLoversize}%
113 \renewcommand*{\L@raise}{\DefaultLraise}%
114 \renewcommand*{\L@ante}{}%
115 \setlength{\L@Findent}{\DefaultFindent}%
116 \setlength{\L@Nindent}{\DefaultNindent}%
117 \setlength{\L@slope}{\DefaultSlope}%
118 \setlength{\L@novskip}{\DiscardVskip}%
119 \ifLettrineImage\L@imagetrue\else\L@imagefalse\fi
120 \ifLettrineOnGrid\L@gridtrue\else\L@gridfalse\fi
121 \ifLettrineRealHeight\L@realhtrue\else\L@realhfalse\fi

```

`\LettrineFont` and `\LettrineFontEPS` both execute `\Lettrine@height` to set `\L@height` which depends on `\L@tbox`. 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 ³.

³Now means before eventually reading the config file.


```

122 \setkeys{L}{#1}%
123 \sbox{\L@tbox}{\LettrineTextFont{\LettrineSecondString}}%
124 \ifL@realh
125   \def\@tempa{#3}
126   \ifx\@tempa\@empty
127     \PackageWarning{lettrine.sty}%
128       {Empty second argument,\MessageBreak
129         ignoring option `realheight';}%
130   \else
131     \sbox{\L@tbox}{\LettrineTextFont{#3}}%
132   \fi
133 \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)...

```

134 \if\DefaultOptionsFile\relax
135 \else
136   \begingroup
137   \InputIfFileExists{\DefaultOptionsFile}%
138     {}%
139     {\PackageWarning{lettrine.sty}%
140       {File \DefaultOptionsFile\space not found}}%
141  }%

```

Gobble the colour commands, just keep the letter argument.

```

142 \def\color##1##{\l@color{##1}}%
143 \let\l@color\@gobbletwo
144 \def\textcolor##1##{\l@textcolor{##1}}%
145 \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).

```

146 \expandafter\ifx\csname l@#2-keys\endcsname\relax
147   \gdef\l@LOKeys{}%
148 \else
149   \xdef\l@LOKeys{\csname l@#2-keys\endcsname}%
150 \fi
151 \endgroup

```

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

```

152 \def\KV@prefix{KV@L@}%
153 \let\@tempc\relax
154 \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.

```

155 \sbox{\L@lbox}{\LettrineFont #2}%
156 \setlength{\LettrineWidth}{\wd\L@lbox}%

```

```

157 \def\KV@prefix{KV@L@}%
158 \let\@tempc\relax
159 \expandafter\KV@do\l@L@Keys,\relax,

```

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

```

160 \setkeys{L}{#1}%
161 \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.

```

162 \ifL@image
163 \sbox{\L@lbox}{\LettrineFontEPS{#2}}%
164 \else
165 \sbox{\L@lbox}{\LettrineFont #2}%
166 \fi

```

Store the initial's final dimensions,

```

167 \setlength{\LettrineWidth}{\wd\L@lbox}%
168 \setlength{\LettrineHeight}{\ht\L@lbox}%
169 \setlength{\LettrineDepth}{\dp\L@lbox}%

```

print some informations about accuracy to the log file,

```

170 \begingroup
171 \def\IeC##1{##1}%
172 \@tempdima=\L@oversize pt\relax
173 \PackageInfo{lettrine.sty}%
174 {Targeted height = \the\L@target@tth\MessageBreak
175 (no accent, loversize=0),\MessageBreak
176 Lettrine height = \the\LettrineHeight\space (#2)%
177 \ifdim\@tempdima=z@\else\space loversize=\L@oversize\fi;%
178 \MessageBreak reported}%
179 \endgroup

```

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

```

180 \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`.

```

181 \setlength{\L@first}{\LettrineHeight}%
182 \setlength{\L@lraise}{\L@raise\L@target@tth}%
183 \addtolength{\L@first}{\L@lraise}%
184 \ifnum\value{L@lines}=1
185 \addtolength{\L@first}{-\ht\L@tbox}%
186 \else
187 \addtolength{\L@first}{-\L@target@tth}%
188 \addtolength{\L@lraise}{-\L@target@dp}%
189 \fi
190 \par

```

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

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

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

```
192 \ifL@grid
193 \tempdima=\baselineskip
194 \tempdimb=\tempdima
195 \advance\tempdimb by \L@novskip
196 \tempcnta=1
197 \loop\ifdim\L@first>\tempdimb
198 \advance\tempcnta by 1
199 \advance\L@first by -\tempdima
200 \repeat
201 \L@first=\tempcnta\baselineskip
202 \fi
203 \vskip\L@first
204 \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.

```
205 \setlength{\L@Pindent}{\wd\L@lbox}%
206 \addtolength{\L@Pindent}{-\L@hang\wd\L@lbox}%
207 \settowidth{\L@first}{\L@ante}%
208 \addtolength{\L@Pindent}{\L@first}%
209 \addtolength{\L@Pindent}{\L@Findent}%
210 \setlength{\L@first}{\linewidth}%
211 \addtolength{\L@first}{-\L@Pindent}%

```

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

```
212 \addtolength{\L@Nindent}{\L@Pindent}%
213 \setlength{\L@next}{\linewidth}%
214 \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).

```
215 \addtolength{\L@Pindent}{\rightmargin}%
216 \addtolength{\L@Nindent}{\rightmargin}%

```

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

```
217 \addtocounter{L@lines}{1}%
218 \addtocounter{L@lines}{\value{L@depth}}%
219 \def\L@parshape{\c@L@lines \the\L@Pindent \the\L@first}%
220 \@tempcnta=\tw@
221 \@whilenum \@tempcnta<\c@L@lines\do{%
222 \edef\L@parshape{\L@parshape \the\L@Nindent \the\L@next}%
223 \addtolength{\L@Nindent}{\L@slope}%
224 \addtolength{\L@next}{-\L@slope}%
225 \advance\@tempcnta\@ne}%
226 \edef\L@parshape{\L@parshape \rightmargin \the\linewidth}%

```

```

227 \noindent
228 \parshape=\L@parshape\relax

```

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

```

229 \smash{\llap{\mbox{\L@ante}\raisebox{\L@lraise}{\usebox{\L@lbox}}}%
230 \hskip \the\L@Findent}}%
231 \usebox{\L@tbox}%
232 }

```

This ends the definition of `\lettrine`.

Load a local config file if present in L^AT_EX's search path.

```

233 \InputIfFileExists{lettrine.cfg}
234 {\typeout{Loading lettrine.cfg}}
235 {\typeout{lettrine.cfg not found, using default values}}

```

3 Configuration file

```

236 %% lettrine.cfg: configuration file for lettrine.sty
237 %%
238 %% If you want to customize lettrine, please do not hack into the
239 %% code, copy this file to your working directory and customize the
240 %% copy as you like.
241 %%
242 %% Uncomment any of these lines and change the parameters' values
243 %% to fit your needs (see lettrine.dtx).
244 %%
245 %%\setcounter{DefaultLines}{2}
246 %%\setcounter{DefaultDepth}{0}
247 %%
248 %% These are decimal numbers:
249 %%\renewcommand*{\DefaultLoversize}{0}
250 %%\renewcommand*{\DefaultLraise}{0}
251 %%\renewcommand*{\DefaultLhang}{0}
252 %%
253 %% These are lengths (don't forget the unit):
254 %%\setlength{\DefaultFindent}{0pt}
255 %%\setlength{\DefaultNindent}{0.5em}
256 %%\setlength{\DefaultSlope}{0mm}
257 %%\setlength{\DiscardVskip}{0.2pt}
258 %%
259 %% These are flags (value=true/false):
260 %%\LettrineImagefalse
261 %%\LettrineOnGridfalse
262 %%\LettrineRealHeightfalse
263 %%
264 %% These are commands (value=string, only height matters):
265 %%\renewcommand*{\LettrineTestString}{EFTZ}
266 %%\renewcommand*{\LettrineSecondString}{x}
267 %% For backward compatibility with versions 1.63 to 1.9:
268 %%\renewcommand*{\LettrineTestString}{ABCDEFGHJKLMNOPQRSTUVWXYZ}

```

```

269 %% For backward compatibility with versions up to 1.62:
270 %%\renewcommand*{\LettrineTestString}{X}
271 %%
272 %% In case you want to set parameters for some letters
273 %% in file `optfile.cfl'
274 %% \renewcommand{\DefaultOptionsFile}{optfile.cfl}

```

4 Change History

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

v2.0		\LettrineFont: \theL@lines changed to \value{L@lines}. Needed for babel-hebrew which redefines \@arabic. 6
\lettrine: Add informations about targeted and actual height of the initial to the .log file. 10	v1.8	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
\LettrineFont: \LettrineTestString changed from ‘ABCDEFGHJKLMNOP- QRSTUVWXYZ’ to ‘EFTZ’ as some capitals like C, G, O, Q or X might be slightly taller (possible optical correction). ... 6		\lettrine: The 0.2pt limit for discarded vskips is now customisable through \DiscardVskip and option ‘novskip’. 10
Change factors 100/100 to 50/200 to reach larger sizes (height up to 600pt). 7		
Store targeted dimensions of the dropped capital (ht, dp, tht) for further use. 6		
v1.9	v1.7	General: New counter to add lines for dropped capitals with positive depth, like Q. 1
General: New customisable string \LettrineSecondString to tune the initial’s height. 3	v1.65	\lettrine: Measure and store the initial’s final dimensions. 10
New keyval option: ‘realheight’ (true/false) and new global flag \ifLettrineRealHeight. 2	v1.64	\lettrine: Remove \$ around \smash and add \relax. Bug pointed out by David Monniaux. Correction by Enrico Gregorio. 12
\lettrine: \theDefaultLines changed to \value{DefaultLines}, same with \theDefaultDepth. Needed for babel-hebrew which redefines \@arabic. Thanks to Ulrike Fischer for providing the fix. 8	v1.63	\LettrineFont: Added command \LettrineTestString which defaults to ‘ABCDEFGHI- JKLMNOPQRSTUVWXYZ’. In previous versions height computations were based on letter ‘X’ which might not exist in some (rare) fonts. Pointed out by Raphaël Pinson. 6
\theL@depth changed to \value{L@depth}. 11	v1.6	General: Add a flag to switch to
Use the second mandatory argument of \lettrine or \LettrineSecondString (which defaults to ‘x’) to compute \L@height. This is controlled by the ‘realheight’ flag. 8		

images in eps or pdf format.		early to \L@Nindent, thus	
Suggested by Bill Jetzer.	2	making \L@next too short by	
Added newif \ifL@grid.	5	\rightmargin.	11
Added newif \ifL@image.	5		
Added newif \ifLettrineImage.	5	v1.3	
\lettrine: Add braces around #3		General: Correct the	
to allow commands taking an		documentation to mention the	
argument (such as		cm-super fonts and the typelec	
\MakeLowercase) in		package by Vladimir Volovich.	4
\LettrineTextFont. Suggested		v1.2	
by Philipp Lehman.	10	General: \newlength changed to	
\LettrineFontEPS: Added		\newdimen, to correct a bug	
\LettrineFontHook to		with seminar.cls (pointed out	
\LettrineFontEPS. This is		by Peter Münster).	5
needed for color options.	7	\lettrine: \baselineskip may be	
v1.5		a rubber length, we convert it	
General: \LettrineOptionsFor		to a dimen.	10
and \LettrineWidth added.	8	\LettrineFont: \baselineskip	
\lettrine: Added reading of an		may be a rubber length, we	
optional config file		convert it to a dimen.	6
\DefaultOptionsFile.	9	v1.1	
v1.4		\lettrine: Add \rightmargin to	
\lettrine: \lettrine still didn't		\L@Pindent for \Lettrine to	
work properly in quote,		work properly in quote,	
quotation, abstract		quotation, abstract	
environments, pointed out by		environments... but do not	
Matthias C. Schmidt.		change \linewidth which is set	
\rightmargin was added too		by these environments.	11