

The `ellipsis` package*

Peter J. Heslin
p.j.heslin@dur.ac.uk

September 28, 2004

1 Introduction

There is an unevenness in the way \LaTeX puts space around ellipses (`\dots`) in text mode. It is a small problem, but it is a serious matter for those who care about such things. This package attempts to fix that bug, and this documentation offers an explanation of the bug and offers advice on how to use ellipses in your text. This document should not be taken as offering guidance on the use of ellipses in mathematical formulas or non-English language text.

The standard definition of `\dots` in \LaTeX takes the inter-word stretch for the current font and puts that amount of space in between three normal dots; it then also adds that amount of space *after* the final dot. The documentation of the \LaTeX code acknowledges that this is a ‘kludge’ in that the interword stretch is being used for a purpose it was not meant for. There is another problem with this definition, however: the extra space after the final dot of the ellipsis. Thus there is always more space after the ellipsis than before it, so that it is not properly centered between the text on either side.

That extra space is there for a good reason. When the ellipsis is followed by another dot, as at the end of a sentence, it is important that all four dots should be evenly spaced, otherwise the final dot is much closer than the others and the result is hideous. The same holds true for commas, exclamation marks and other punctuation on the baseline. So the extra space is necessary when an ellipsis is followed by certain punctuation characters, and the bug consists in the fact that \LaTeX always adds it, regardless of what text follows.

This package implements a simple fix. It redefines the `\dots` and `\textellipsis` commands so that they can look ahead at the next character and change their behavior accordingly: if the next character is one of `,;:?!` the extra space is added, if not, no extra space is added (if you load a package that makes any of these characters active, you may want to reload this list; see the section below on `\ellipsis punctuation` and compatibility). This particular list of punctuation marks was not chosen arbitrarily: it includes all marks with a dot on the baseline (or a comma, which is like a dot with a tail). These marks produce a series of four dots which must be spaced evenly – no other punctuation mark would normally benefit from having the extra space added.

*This document corresponds to `ellipsis` v1.6, dated 2004/9/28. Many thanks to Frank Mittelbach, who made numerous suggestions that greatly improved this package.

The solution can be brittle – for example, if you write `\dots{}`. then the extra space will *not* be added, since the braces come between the command and the dot; the result will be very ugly. The solution to this is to enter ellipses carefully and consistently in your source text. If you use ellipses as recommended below, then the potential problem of evenly spacing an ellipsis that comes immediately after a baseline punctuation mark does not arise, since that combination of characters will not normally be used.

There is another package, `lips.sty`, that addresses the problem of text ellipses in \LaTeX . It strictly follows the advice of the *Chicago Manual of Style* in putting full word spaces between the dots of the ellipsis, and does not provide the possibility of more a tightly set ellipsis like the normal \LaTeX default. It imposes its own set of rules for the spacing before and after an ellipsis. Many typographers, however, firmly reject the Chicago style of setting ellipses, and rightly so; see Jan Tschichold, *The Form of the Book: Essays on the Morality of Good Design* (Hartley & Marks, 1991), pp 130f, or Robert Bringhurst, *The Elements of Typographic Style* (2nd ed, `chicago` Hartley & Marks, 1997) pp 82f. If you nevertheless want or have to use the Chicago-style, widely spaced ellipsis with this package, you can pass it the option `chicago`, like so: `\usepackage[chicago]{ellipsis}`. If, on the other hand, you want to adhere to the full recommendations of the Chicago manual, then you should probably use `lips.sty` instead of this package. `Ellipsis.sty` also has an `m1a` package option, modeled on the same feature of `lips.sty`, which automatically puts square brackets around all ellipses.

A different solution to the general problem would be to install a font which includes a precomposed ellipsis glyph, and to redefine `\dots` simply to insert this character. In that case, you do not need this package at all, but you should make sure that the ellipsis character kerns properly, especially with following `.,;!?`

2 Usage

Install the package and put `\usepackage{ellipsis}` in your preamble. As noted above, you should be consistent in the way you enter ellipses in your text. For English text, Bringhurst (loc. cit.) recommends putting a space before and after an ellipsis that appears between two words, but no space before an ellipsis that appears before a punctuation mark such as a period, comma, etc. Then there is the question of breaking or non-breaking space. I think it is odd to find an ellipsis at the beginning of a line, so normally I would enter text like this: `uh~\dots\ oh`. If setting text in narrow columns you may prefer to allow line breaks before the ellipsis as well as after. Before punctuation, you would enter ellipses without a space before or after, like so: `one, two, three\dots, ten\dots`. Note, however, that some publishers do not like to see an ellipsis combined with a period or comma, and would rather a simple ellipsis at the end of a sentence and so forth.

One problem with using the `\dots` command is that it does not take an argument. So there is no brace to terminate it, and if you want a space to follow the ellipsis, you need to take care that it does not disappear, and so to write it like this: `\dots\~` or `{\dots}\~` or `\dots{ }\~`. If you forget to do this, the space will disappear, gobbled up by the macro. I frequently find myself making this mistake, so I have added another option to help with it. If you pass the option `xspace` to `ellipsis.sty`, an `\xspace` macro will be added after every `\dots` macro. See the *L^AT_EX Companion* for full information on that package, but in short, it

adds a space except when followed by certain punctuation characters (a superset of the `\ellipsis punctuation` list mentioned above). So you automatically get space after the ellipsis, unless it is immediately followed by punctuation, even if you write it like this: `one~\dots two`. If there are a few places where you don't want a space after `\dots`, then write it like this: `\dots{}`, and the space will be suppressed.

`\ellipsisgap`

You may wish to redefine the command `\ellipsisgap`, which is the space between the dots of the ellipsis. If you do this, you must do it in your preamble, *after* the `\usepackage` line. The default value is the interword stretch of the current font, which is the normal L^AT_EX definition. If you are using a font other than Computer Modern and if you are obsessive, you might wish to examine the ellipsis that the font designer included in your font and recreate it by defining the `\ellipsisgap` to an appropriate value, like so:

```
\usepackage{ellipsis}
This value seems right for the native ellipsis in Adobe Caslon:
\renewcommand{\ellipsisgap}{0.1em}
```

You should probably define this in font-size dependent units, such as the em. See Bringhurst (loc. cit.) for further thoughts on the construction of ellipses.

3 Compatibility

It was mentioned above that if certain punctuation characters are made active, it will interfere with the ability of this package to recognize them. One important example of that practice is in the French option of Babel, which makes certain 'double punctuation' characters active, in order to put a bit of space before them. In this particular case, however, you do *not* want to redefine the `\ellipsis punctuation` list to make this package aware of these characters. If you were to do so, extra kerning would be added after an ellipsis in addition to thin space before the following punctuation, resulting in about twice as much space as needed. If you leave the default value of `\ellipsis punctuation`, then these active characters will not be recognized, and no extra kern will be added. Instead, the ellipsis will be separated from any following double punctuation by a thin space. This may not be quite the same as the space between dots of the ellipsis, but it may be close enough that no one will notice the difference. In any case, that's what French Babel does.

Whether Babel is here following some sophisticated French typographical rule, or it's just a fudge, I don't know. Because Babel French removes the extra space from after `\dots`, but only adds space before 'double punctuation', there is no space between an ellipsis and 'single punctuation', which looks ugly to my (non-French) eyes. In these cases, using `ellipsis.sty` will add the extra space whereas French Babel on its own does not. This may be the wrong thing to do; if you are typesetting a text in French, you should familiarize yourself with the relevant norms, and possibly refrain from using this package. I have not familiarized myself with the rules for typesetting ellipses in languages other than English. If you are loading the French option of Babel because you are quoting French text in a document whose main language is not French, then you should not worry about these issues. Because French Babel wants to redefine `\dots`, you must load `ellipsis.sty` *after* Babel if you are using the French option (even if you have no French text in your document).

The Spanish option of Babel has its own way of setting an ellipsis, but it uses a different command (`\dots`), so that doesn't interfere with the working of this package.

4 Implementation

<code>\ellipsisgap</code>	Set the amount of space to put between the dots of the ellipse. Defaults to the standard L ^A T _E X amount. <pre> 1 \newcommand{\ellipsisgap}{\fontdimen3\font} 2 \DeclareOption{chicago}{\renewcommand{\ellipsisgap}{\fontdimen2\font}}</pre>
<code>\ellipsis@before</code> <code>\ellipsis@after</code>	We provide the hooks <code>\ellipsis@before</code> and <code>\ellipsis@after</code> to allow the production of automatically bracketed ellipses. <pre> mla The mla package option sets these so as to produce ellipses like this: [...] If you do this, you never want the extra space after the ellipsis, so we set \ellipsis@alwayscentertrue 3 \newcommand{\ellipsis@before}{} 4 \newcommand{\ellipsis@after}{} 5 \newif\ifellipsis@alwayscenter 6 \ellipsis@alwayscenterfalse 7 \DeclareOption{mla}{% 8 \renewcommand{\ellipsis@before}{\kern\ellipsisgap}% 9 \renewcommand{\ellipsis@after}{\kern\ellipsisgap }} 10 \ellipsis@alwayscentertrue</pre>
<code>\ellipsis@xspace</code>	Append <code>\xspace</code> if the <code>xspace</code> option is set. <pre> 11 \newcommand\ellipsis@xspace{} 12 \DeclareOption{xspace}{% 13 \renewcommand{\ellipsis@xspace}{\xspace}} 14 \ProcessOptions\relax 15 \RequirePackage{xspace}</pre>
<code>\ellipsis@default</code>	This is the L ^A T _E X default definition, which is necessary to use when punctuation such as <code>.,:;!?</code> follows. <pre> 16 \newcommand{\ellipsis@default}{% 17 \ellipsis@before 18 .\kern\ellipsisgap 19 .\kern\ellipsisgap 20 .\kern\ellipsisgap 21 \ellipsis@after\relax}</pre>
<code>\ellipsis@centered</code>	This is our new ellipsis <i>without</i> the extra space after it. <pre> 22 \newcommand{\ellipsis@centered}{% 23 \ellipsis@before 24 .\kern\ellipsisgap 25 .\kern\ellipsisgap 26 .\ellipsis@after\relax}</pre>
<code>\ellipsis@punctuation</code>	Here we define the list of punctuation marks before which we want to put <code>\ellipsis@spacing</code> space. This may be redefined by the user if desired. If you load a package that changes the <code>\catcode</code> of a character in this list, such as a language package that makes one of them active, and you want the extra

kerning to be added in front of those characters, then you ‘must then explicitly reset the list. Otherwise the changed character will no longer be recognized.’ (quote from `lftncmd.dtx`) In such a case, just repeat the line below, substituting `\renewcommand` for `\newcommand`.

```
27 \newcommand\ellipsisispunctuation{.,:;!/?}
```

`\textellipsis` This is the new definition for an ellipsis which looks ahead: if the next char is in `\ellipsisispunctuation` use `\default@ellipsis`, else use our new `\center@ellipsis`.

This is the auxiliary code that scans through the list of punctuation.

```
28 \def\ellipsis@scan{\expandafter\ellipsis@scan@aux\ellipsisispunctuation\ellipsis@delim}
29 \def\ellipsis@scan@aux #1#2\ellipsis@delim{%
30   \let\ellipsis@one=#1%   the first char
31   \def\ellipsis@two{#2}%  the remainder of the string
32   \ifx\ellipsis@token\ellipsis@one
33     \ellipsis@default
34   \else
35     \ifx\ellipsis@two\empty
36       \ellipsis@centered
37     \else
38       \ellipsis@scan@aux #2\ellipsis@delim
39     \fi
40 \fi}
```

Here is the macro that looks ahead at the next token, put it in `\ellipsis@token`, and invokes the code to scan for it in the punctuation list.

```
41 \renewcommand{\textellipsis}{\futurelet\ellipsis@token\@textellipsis}
42 \def\@textellipsis{%
43   \ifellipsis@alwayscenter\ellipsis@centered\else
44     \ellipsis@scan%
45   \fi\ellipsis@xspace}%

```

`\dots` Here we redefine the standard L^AT_EX command to use our new definition. (the `\expandafter` needs to be added to make lookahead work). The only reason this must be done at the beginning of the document is that the French option of Babel unpleasantly redefines `\dots` at the beginning of the document, and so we have to override it again – and this is true even if you never use any French text in your document. We need to throw an error in the case where French Babel is loaded after us, because its `AtBeginDocument` hook is about to clobber our definition of `\dots`.

```
46 \ifx\bbf@frenhdots\@undefined\else\def\ellipsis@frenchloaded{\relax}\fi
47 \AtBeginDocument{%
48   \ifx\ellipsis@frenchloaded\@undefined
49     \ifx\bbf@frenhdots\@undefined\else
50       \PackageError{ellipsis}{Babel French loaded after ellipsis.sty}%
51       {If you load Babel with the French option, do it before ellipsis.sty}%
52     \fi
53   \fi
54   \DeclareRobustCommand{\dots}{%
55     \ifmmode\mathellipsis\else\expandafter\textellipsis\fi}}
```

`\midwordellipsis` An extra command: this may be useful for the rare time when you want an ellipsis in the very middle of a word or whenever you just want a small bit of space (the intra-ellipsis spacing) before and after the ellipsis.

```
56 \DeclareRobustCommand{\midwordellipsis}{%  
57   \kern\ellipsisgap  
58   .\kern\ellipsisgap  
59   .\kern\ellipsisgap  
60   .\kern\ellipsisgap\relax}
```