

# The `abstract` package\*

Peter Wilson<sup>†</sup>  
Catholic University of America  
Now at `peter.r.wilson@boeing.com`

2004/04/16

## Abstract

The `abstract` package provides control over the typesetting of the `abstract` environment, and especially provides for a one-column abstract in a two-column paper.

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>The <code>abstract</code> package</b>	<b>2</b>
2.1	Options . . . . .	2
2.2	Commands and environments . . . . .	3
<b>3</b>	<b>The package code</b>	<b>4</b>
3.1	Preliminaries . . . . .	4
3.2	Abstracts . . . . .	5

## 1 Introduction

Questions about how to have a one-column abstract in a two-column paper seem to pop up fairly regularly on the `comp.text.tex` newsgroup. While an answer based on responses on `ctt` is provided in the FAQ, the `abstract` package provides a more author-friendly means of accomplishing this. Further, additional controls are provided for the typesetting of the `abstract` environment in general.

This manual is typeset according to the conventions of the  $\text{\LaTeX}$  `DOCSTRIP` utility which enables the automatic extraction of the  $\text{\LaTeX}$  macro source files [GMS94].

Section 2 describes the usage of the `abstract` package and commented source code is in Section 3.

---

\*This file (`abstract.dtx`) has version number v1.1a, last revised 2004/04/16.

<sup>†</sup>After May 2004 at: `pandgwilson at earthlink dot net`

## 2 The abstract package

The typeset format of the `abstract` in a report or article class<sup>1</sup> document depends on the class options. The formats are:

- `titlepage` class option: The abstract heading (i.e., value of `\abstractname`) is typeset centered in a bold font; the text is set in the normal font and to the normal width.
- `twocolumn` class option: The abstract heading is typeset like an unnumbered section; the text is set in the normal font and to the normal width (of a single column).
- Default (neither of the above class options): The abstract heading is typeset centered in a small bold font; the text is set in a small font and indented like the `quotation` environment.

Note that the `titlepage` option takes precedence over the `twocolumn` option.

The `abstract` package provides handles to modify the typesetting of the `abstract`.

### 2.1 Options

The `abstract` package takes the following options.

- `original`. In a `twocolumn` document the package default is to use the default abstract style (i.e., centered heading and small text). This option sets the typeset format of the abstract to match that of a standard two column document (i.e., section heading and normal sized text).
- `addtotoc`. The abstract title (the value of `\abstractname`) is added to the Table of Contents.
- `number`. The abstract title will be typeset as a numbered `\chapter` or as a `\section`, depending on whether the document class supports chapters or not.
- `runin`. The abstract title is typeset as a run-in<sup>2</sup> heading. Note that the `number` option is ignored if the `runin` option is used.

When the `number` option is used with a class that provides chapters, the usual `\chapter` command is used for typesetting the title. If you do not want the word ‘Chapter’ (or its equivalent) to appear, you can use the `anonchp` package to suppress this.

---

<sup>1</sup>The `abstract` environment is not available for the `book` class.

<sup>2</sup>This option was suggested by R. Chandrasekhar (`chandra@ee.uwa.edu.au`).

## 2.2 Commands and environments

The usual advice about creating a one-column `abstract` in a two-column document is to write code like this:

```
\documentclass[twocolumn...]{...}
...
\twocolumn[
  \begin{@twocolumnfalse}
    \maketitle           % need full-width title
    \begin{abstract}
      abstract text...
    \end{abstract}
  \end{@twocolumnfalse}
]
... hand make footnotes for any \thanks commands
...
```

`onecolabstract`      With the `abstract` package, instead you do:

```
\saythanks
\documentclass[twocolumn...]{...}
\usepackage{abstract}
...
\twocolumn[
  \maketitle           % need full-width title
  \begin{onecolabstract}
    abstract text...
  \end{onecolabstract}
]
\saythanks % typesets any \thanks commands
...
```

If you want, you can use the `onecolabstract` environment in place of the `abstract` environment — it doesn't have to be within the optional argument of the `\twocolumn` command. In fact, `onecolabstract` internally uses `abstract` for the typesetting.

`\abstractnamefont`      These two commands can be redefined to change the fonts used for typesetting  
`\abstracttextfont`    the heading (defined via `\abstractname`) of the `abstract` environment and the  
font for typesetting the text of the abstract, respectively. The default definitions  
for these are designed to mimic the traditional `abstract` typesetting. Different  
values are used depending on whether the document uses the `titlepage` and/or  
`twocolumn` options. For example, in a non-titlepage one-column paper, their definitions are:

```
\newcommand{\abstractnamefont}{\normalfont\small\bfseries}
\newcommand{\abstracttextfont}{\normalfont\small}
```

<code>\absleftindent</code>	The amended version of <code>abstract</code> uses a list environment for typesetting the text. These four lengths can be changed (via <code>\setlength</code> ) to adjust the left and right margins, the paragraph indentation, and the vertical skip between paragraphs in this environment. The default values depend on the document class options in effect.
<code>\absrightindent</code>	
<code>\absparindent</code>	
<code>\absparsep</code>	
<code>\abslabeldelim</code>	If the <code>runin</code> option is used, the heading is typeset as a run-in heading. That is, it is the first piece of text on the first line of the abstract text. The $\langle text \rangle$ as specified by <code>\abslabeldelim{\langle text \rangle}</code> is typeset immediately after the heading. By default it is defined to do nothing, but if you wanted, for example, the <code>\abstractname</code> to be followed by a colon and some extra space you could specify <code>\abslabeldelim{:\quad}</code> .
<code>\absnamepos</code>	If the <code>runin</code> option is not used then the heading is typeset in its own environment. <code>\absnamepos</code> specifies the name of the environment in which the abstract name is typeset. It can be redefined to be one of <code>flushleft</code> , <code>center</code> , or <code>flushright</code> to give a left, centered or right aligned heading; or to any other appropriate environment which is supported by the document class or used packages.
<code>\abstitleskip</code>	With the <code>runin</code> option a space of length <code>\abstitleskip</code> is typeset before the heading. For example, if <code>\absparindent</code> is non-zero, then: <code>\setlength{\abstitleskip}{-\absparindent}</code> will typeset the heading flush left. Without the <code>runin</code> option <code>\abstitleskip</code> is additional vertical space (either positive or negative) that is inserted between the abstract name and the text of the abstract.
<code>\appendiargdef</code>	The (internal) command <code>\appendiargdef{\macro}{\stuff}</code> will append $\langle stuff \rangle$ at the end of the current definition of $\langle macro \rangle$ , where $\langle macro \rangle$ is the name of a macro (including the backslash) which takes a single argument. For example the following are two equivalent definitions of <code>\mymacro</code> :

```

\newcommand{\mymacro}[1]{#1 is an artist}
\appendiargdef{\mymacro}{ in spite of being colour blind}
% or
\newcommand{\mymacro}[1]{#1 is an artist in spite of being colour blind}

```

### 3 The package code

To try and avoid name clashes, all the internal commands include the string `@bs`.

#### 3.1 Preliminaries

Announce the name and version of the package, which requires L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.

```

1 \*usc
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{abstract}[2001/02/11 v1.1 configurable abstracts]
4

```

The following `\if...` commands are for implementing various options.

```

\if@bsonecol
\ifadd@bstotoc 5 \newif\if@bsonecol
\ifnumber@bs 6 \@bsonecoltrue
\if@bsrunin 7 \newif\ifadd@bstotoc
8 \add@bstotocfalse
9 \newif\ifnumber@bs
10 \number@bsfalse
11 \newif\if@bsrunin
12 \@bsruninfalse
13

```

Now declare and process the options.

```

14 \DeclareOption{original}{\@bsonecolfalse}
15 \DeclareOption{addtotoc}{\add@bstotoctrue}
16 \DeclareOption{number}{\number@bstrue}
17 \DeclareOption{runin}{\@bsrunintrue}
18 \ProcessOptions\relax

```

Make sure that the number option is disabled if runin is used (otherwise the addtotoc option may not work as expected).

```

19 \if@bsrunin\number@bsfalse\fi
20

```

## 3.2 Abstracts

The original definition of the `abstract` environment given in `ltclasses.dtx` is:

```

\if@titlepage
\renewenvironment{abstract}{%
\titlepage
\null\vfil
\@beginparpenalty\@lowpenalty
\begin{center}%
\bfseries \abstractname
\@endparpenalty\@M
\end{center}}%
{\par\vfil\null\endtitlepage}
\else
\renewenvironment{abstract}{%
\if@twocolumn
\section*{\abstractname}%
\else
\small
\begin{center}%
{\bfseries \abstractname\vspace{-.5em}\vspace{\z@}}%
\end{center}%
\quotation
\fi}
{\if@twocolumn\else\endquotation\fi}
\fi

```

The next set of macros comprise the abstract package reimplementa-  
tion of the abstract environment.

```

\abstractnamefont These two macros are for specifying the fonts for typesetting the abstract's title
\abstracttextfont and text. They are initialised for the default case (i.e., no class options).
21 \newcommand{\abstractnamefont}{\normalfont\small\bfseries}
22 \newcommand{\abstracttextfont}{\normalfont\small}
23

\absnamepos \absnamepos specifies the environment in which the abstract's title name will be
\abstitlekip typeset, and the length \abstitlekip is an adjustment to the vertical space
between the title and the abstract's text. These are initialised for the default
case.
24 \newcommand{\absnamepos}{center}
25 \newlength{\abstitlekip} \setlength{\abstitlekip}{-0.5em}

\absleftindent The abstract's text is typeset as a single item list, called @bstr@ctlist. These
\absrightindent lengths set the left and right margin indents, the paragraph indentation, and the
\absparindent inter-paragraph vertical space. Their initial values are all class option-dependent.
\absparsep
26 \newlength{\absleftindent}
27 \newlength{\absrightindent}
28 \newlength{\absparindent}
29 \newlength{\absparsep}
30

\abslabeldelim The contents of \abslabeldelim are typeset after a run-in heading.
\@bslabeldelim 31 \newcommand{\abslabeldelim}[1]{\def\@bslabeldelim{#1}}
32 \abslabeldelim{}

\@bsrunintitle This macro typeset the run-in heading.
33 \newcommand{\@bsrunintitle}{%
34 \hspace*{\abstitlekip}{\abstractnamefont\abstractname\@bslabeldelim}}
35

```

Now set all the class option-dependent values.

```

36 \if@titlepage
Values for the titlepage class option. Note that this option takes precedence over
any other option.
37 \setlength{\absleftindent}{\z@}
38 \renewcommand{\abstractnamefont}{\normalfont\bfseries}
39 \renewcommand{\abstracttextfont}{\normalfont}
40 \setlength{\abstitlekip}{0em}
41 \else
42 \if@twocolumn

```

Values for the twocolumn class option.

```
43 \if@bsonecol
44 \setlength{\absleftindent}{\leftmargin}
45 \else
46 \setlength{\absleftindent}{\z@}
47 \renewcommand{\abstractnamefont}{\normalfont\Large\bfseries}
48 \renewcommand{\abstracttextfont}{\normalfont}
49 \renewcommand{\absnamepos}{flushleft}
50 \setlength{\abstitlekip}{0em}
51 \fi
52 \else
```

Values for the default, if not already initialised.

```
53 \setlength{\absleftindent}{\leftmargin}
54 \fi
55 \fi
```

Finally, values that apply for all cases.

```
56 \setlength{\absrightindent}{\absleftindent}
57 \AtBeginDocument{\setlength{\absparindent}{\parindent}
58 \setlength{\absparsep}{\parskip}}
59
```

`@bstr@ctlist` The abstract's text is typeset within the `@bstr@ctlist` list environment.

```
60 \newenvironment{@bstr@ctlist}{%
61 \list}{%
62 \topsep \z@
63 \partopsep \z@
64 \listparindent \absparindent
65 \itemindent \listparindent
66 \leftmargin \absleftindent
67 \rightmargin \absrightindent
68 \parsep \absparsep}%
69 \item\relax}
70 {\endlist}
71
```

`\put@bsintoc` This macro adds the abstract's title to the ToC. It does nothing if the abstract is being numbered.

```
72 \newcommand{\put@bsintoc}{%
73 \ifadd@bstotoc
74 \ifnumber@bs\else
75 \ifundefined{chapter}{\addcontentsline{toc}{section}{\abstractname}}{%
76 \addcontentsline{toc}{chapter}{\abstractname}}
77 \fi
78 \fi}
79
```

`\num@bs` This macro generates a numbered abstract heading.

```
80 \newcommand{\num@bs}{%
```

```

81 \ifundefined{chapter}{\section{\abstractname}}{%
82     \chapter{\abstractname}}
83 }
84

```

**abstract** At last we are in position to redefine the **abstract** environment. This follows very much along the lines of the original definition, but with macros inserted at strategic points.

```
85 \if@titlepage
```

For the titlepage option.

```

86 \renewenvironment{abstract}{%
87     \titlepage
88     \null\vfil
89     \@beginparpenalty\@lowpenalty
90     \if@bsrunin\else

```

If the runin option is not used: number the title, if called for, otherwise typeset the title in the specified (via `\absnamepos`) environment and adjust the following vertical spacing.

```

91     \ifnumber@bs \num@bs \else
92     \begin{\absnamepos}%
93     \abstractnamefont \abstractname
94     \@endparpenalty\@M
95     \end\absnamepos%
96     \vspace{\abstitleskip}%
97     \fi
98     \fi

```

Try adding the title to the ToC, then start the environment for typesetting the text. If the runin option is used, typeset the heading at the start of the environment.

```

99     \put@bsintoc%
100     \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%

```

Finish the **abstract** environment.

```

101     {\par\end{@bstr@ctlist}\vfil\null\endtitlepage}
102 \else

```

The `twocolumn` option and the default case. These use the same code as any style differences are embedded in the new macros.

```

103 \renewenvironment{abstract}{%
104     \if@bsrunin\else
105     \ifnumber@bs \num@bs \else
106     \begin{\absnamepos}\abstractnamefont\abstractname\end\absnamepos%
107     \vspace{\abstitleskip}%
108     \fi
109     \fi
110     \put@bsintoc%
111     \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
112     {\par\end{@bstr@ctlist}}
113 \fi
114

```

`onecolabstract` An environment for typesetting a single column abstract, particularly as the optional argument to the `\twocolumn` command.

```
115 \newenvironment{onecolabstract}{%
116 \begin{@twocolumnfalse}\begin{abstract}}{%
117 \end{abstract}\end{@twocolumnfalse}}
118
```

`\appendiargdef` `\appendiargdef{\macro}{Additional stuff}` will add `Additional stuff` at the end of the definition of `\macro`, where `\macro` is a macro that has one argument.<sup>3</sup>

```
119 \providecommand{\appendiargdef}[2]{\begingroup
120 \toks@ \expandafter{#1{##1}#2}%
121 \edef\@bsx{\endgroup \def\noexpand#1###1{\the\toks@}}%
122 \@bsx}
123
```

`\thanks` We have to keep the contents of the `\thanks` commands as normally these are emptied by the `\maketitle` command. I do this by extending the definition of the `\thanks` (from `ltsect.dtx`) command, so that `\@bs@thanks` has a copy of the contents of `\@thanks`.

```
124 \appendiargdef{\thanks}{%
125 \protected@xdef\@bs@thanks{\@bs@thanks
126 \protect\footnotetext[\the\c@footnote]{#1}}%
127 }
128 \let\@bs@thanks\@empty
129
```

`\saythanks` This macro typesets any `\thanks` commands *after* using `onecolabstract`.

```
130 \newcommand{\saythanks}{\begingroup
131 \renewcommand{\thefootnote}{\fnsymbol{footnote}}\@bs@thanks
132 \endgroup\global\let\@bs@thanks\@empty}
133
```

The end of this package.

```
134 </usc>
```

## References

[GMS94] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley Publishing Company, 1994.

---

<sup>3</sup>Code suggestions for this were made by Michael Downes ([epsmjd@ams.org](mailto:epsmjd@ams.org)) and Heiko Oberdiek ([oberdiek@ruf.uni-freiburg.de](mailto:oberdiek@ruf.uni-freiburg.de)) on `ctt`.

# Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

	<b>Symbols</b>		
<code>\@beginparpenalty</code>	. 89	<code>\abstracttextfont</code>	<i>3</i> , <u>21</u> , 39, 48, 100, 111
<code>\@bs@thanks</code>	<u>124</u> , 131, 132	<code>\add@bstotocfalse</code>	.. 8
<code>\@bslabeldelim</code>	.. <u>31</u> , 34	<code>\add@bstotoctrue</code>	.. 15
<code>\@bsonecolfalse</code>	... 14	<code>\addcontentsline</code>	75, 76
<code>\@bsonecoltrue</code>	..... 6	<code>\appendiargdef</code>	.....
<code>\@bsrunifalse</code>	.... 12		..... <i>4</i> , <u>119</u> , 124
<code>\@bsrunintitle</code>	.....	<code>\AtBeginDocument</code>	.. 57
	..... <u>33</u> , 100, 111		
<code>\@bsrunintrue</code>	..... 17	<b>C</b>	
<code>@bstr@ctlist</code> (environ-	..... <u>60</u>	<code>\c@footnote</code>	..... 126
ment)	.....	<code>\chapter</code>	..... 82
<code>\@bsx</code>	..... 121, 122		
<code>\@endparpenalty</code>	... 94	<b>E</b>	
<code>\@ifundefined</code>	... 75, 81	<code>\edef</code>	..... 121
<code>\@lowpenalty</code>	..... 89	<code>\endtitlepage</code>	..... 101
		environments:	
<b>A</b>		<code>@bstr@ctlist</code>	... <u>60</u>
<code>\abslabeldelim</code>	... <i>3</i> , <u>31</u>	<code>abstract</code>	..... <u>85</u>
<code>\absleftindent</code>	.....	<code>oncolabstract</code>	<i>3</i> , <u>115</u>
	..... <i>3</i> , <u>26</u> , 37,		
	44, 46, 53, 56, 66	<b>F</b>	
<code>\absnamepos</code>	..... <i>4</i> ,	<code>\fnsymbol</code>	..... 131
	<u>24</u> , 49, 92, 95, 106	<code>\footnotetext</code>	..... 126
<code>\absparindent</code>	.....		
	..... <i>3</i> , <u>26</u> , 57, 64	<b>I</b>	
<code>\absparsep</code>	<i>3</i> , <u>26</u> , 58, 68	<code>\if@bsonecol</code>	..... <u>5</u> , 43
<code>\absrightindent</code>	...	<code>\if@bsrunin</code>	..... <u>5</u> ,
	..... <i>3</i> , <u>26</u> , 56, 67		19, 90, 100, 104, 111
<code>\abstitlekip</code>	<i>4</i> , <u>24</u> ,	<code>\if@titlepage</code>	... 36, 85
	34, 40, 50, 96, 107	<code>\if@twocolumn</code>	..... 42
<code>abstract</code> (environ-	..... <u>85</u>	<code>\ifadd@bstotoc</code>	... <u>5</u> , 73
ment)	.....	<code>\ifnumber@bs</code>	.....
<code>\abstractname</code>	34, 75,		..... <u>5</u> , 74, 91, 105
	76, 81, 82, 93, 106	<code>\item</code>	..... 69
<code>\abstractnamefont</code>	.....	<code>\itemindent</code>	..... 65
	..... <i>3</i> , <u>21</u> ,		
	34, 38, 47, 93, 106	<b>L</b>	
		<code>\Large</code>	..... 47
		<code>\leftmargin</code>	.. 44, 53, 66
		<code>\listparindent</code>	.. 64, 65
		<b>N</b>	
		<code>\newif</code>	..... 5, 7, 9, 11
		<code>\noexpand</code>	..... 121
		<code>\normalfont</code>	..... 21,
			22, 38, 39, 47, 48
		<code>\num@bs</code>	... <u>80</u> , 91, 105
		<code>\number@bsfalse</code>	. 10, 19
		<code>\number@bstrue</code>	..... 16
		<b>O</b>	
		<code>oncolabstract</code> (envi-	.....
		ronment)	.. <i>3</i> , <u>115</u>
		<b>P</b>	
		<code>\parindent</code>	..... 57
		<code>\parsep</code>	..... 68
		<code>\parskip</code>	..... 58
		<code>\partopsep</code>	..... 63
		<code>\protected@xdef</code>	... 125
		<code>\providecommand</code>	... 119
		<code>\ProvidesPackage</code>	... 3
		<code>\put@bsintoc</code>	<u>72</u> , 99, 110
		<b>R</b>	
		<code>\renewenvironment</code>	.....
			..... 86, 103
		<b>S</b>	
		<code>\saythanks</code>	..... <i>3</i> , <u>130</u>
		<code>\section</code>	..... 81
		<code>\small</code>	..... 21, 22
		<b>T</b>	
		<code>\thanks</code>	..... <u>124</u>
		<code>\thefootnote</code>	..... 131
		<code>\titlepage</code>	..... 87
		<code>\toks@</code>	..... 120, 121
		<code>\topsep</code>	..... 62