

The `abstract` package*

Peter Wilson
Catholic University of America
Now at `peter.r.wilson@boeing.com`

2000/06/11

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

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

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 L^AT_EX DOCSTRIP utility which enables the automatic extraction of the L^AT_EX macro source files [GMS94].

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

*This file has version number v1.0, last revised 2000/06/11.

2 The abstract package

The typeset format of the `abstract` in a `report` or `article` class¹ 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.

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.

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
```

¹The `abstract` environment is not available for the `book` class.

```

        \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 defini-
tions are:

```

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

```

`\absnamepos` `\absnamepos` specifies the name of the environment in which the abstract name
`\abstitleskip` is typeset. It can be redefined to be one of `flushleft`, `center`, or `flushright`
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.

The length `\abstitleskip` is additional vertical space (either positive or negative) that is inserted between the abstract name and the text of the abstract.

`\absleftindent` The amended version of `abstract` uses a list environment for typesetting the
`\absrightindent` text. These four lengths can be changed (via `\setlength`) to adjust the left and
`\absparindent` right margins, the paragraph indentation, and the vertical skip between para-
`\absparsep` graphs. The default values depend on the document options in effect.

`\appendiargdef` The (internal) command `\appendiargdef{<macro>}{<stuff>}` will append
`<stuff>` at the end of the current definition of `<macro>`, where `<macro>` is the name

of a macro (including the backslash) which takes a single argument. For example the following are two equivalent definitions of `\mymacro`:

```
\newcommand{\mymacro}[1]{#1 is an artist}
\appendargdef{\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^AT_EX 2_ε.

```
1 (*usc)
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{abstract}[2000/06/11 v1.0 configurable abstracts]
4
```

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

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

Now declare and process the options.

```
12 \DeclareOption{original}{\@bsonecolfalse}
13 \DeclareOption{addtotoc}{\add@bstotoctrue}
14 \DeclareOption{number}{\number@bstrue}
15 \ProcessOptions\relax
16
```

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 reimplementation of the `abstract` environment.

`\abstractnamefont` `\abstracttextfont` These two macros are for specifying the fonts for typesetting the abstract's title and text. They are initialised for the default case (i.e., no class options).

```

17 \newcommand{\abstractnamefont}{\normalfont\small\bfseries}
18 \newcommand{\abstracttextfont}{\normalfont\small}
19

```

`\absnamepos` `\abstitlekip` `\absnamepos` specifies the environment in which the abstract's title name will be 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.

```

20 \newcommand{\absnamepos}{center}
21 \newlength{\abstitlekip} \setlength{\abstitlekip}{-0.5em}

```

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

```

22 \newlength{\absleftindent}
23 \newlength{\absrightindent}
24 \newlength{\absparindent}
25 \newlength{\absparsep}
26

```

Now set all the class option-dependent values.

```

27 \if@titlepage

```

Values for the `titlepage` class option. Note that this option takes precedence over any other option.

```

28 \setlength{\absleftindent}{\z@}
29 \renewcommand{\abstractnamefont}{\normalfont\bfseries}
30 \renewcommand{\abstracttextfont}{\normalfont}
31 \setlength{\abstitlekip}{0em}
32 \else
33 \if@twocolumn

```

Values for the twocolumn class option.

```
34 \if@bsonecol
35 \setlength{\absleftindent}{\leftmargin}
36 \else
37 \setlength{\absleftindent}{\z@}
38 \renewcommand{\abstractnamefont}{\normalfont\Large\bfseries}
39 \renewcommand{\abstracttextfont}{\normalfont}
40 \renewcommand{\absnamepos}{flushleft}
41 \setlength{\abstitleskip}{0em}
42 \fi
43 \else
```

Values for the default, if not already initialised.

```
44 \setlength{\absleftindent}{\leftmargin}
45 \fi
46 \fi
```

Finally, values that apply for all cases.

```
47 \setlength{\absrightindent}{\absleftindent}
48 \AtBeginDocument{\setlength{\absparindent}{\parindent}
49 \setlength{\absparsep}{\parskip}}
50
```

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

```
51 \newenvironment{@bstr@ctlist}{%
52 \list{}{%
53 \topsep \z@
54 \partopsep \z@
55 \listparindent \absparindent
56 \itemindent \listparindent
57 \leftmargin \absleftindent
58 \rightmargin \absrightindent
59 \parsep \absparsep}%
60 \item\relax}
61 {\endlist}
62
```

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

```
63 \newcommand{\put@bsintoc}{%
64 \ifadd@bstotoc
65 \ifnumber@bs\else
66 \ifundefined{chapter}{\addcontentsline{toc}{section}{\abstractname}}{%
67 \addcontentsline{toc}{chapter}{\abstractname}}
68 \fi
69 \fi}
70
```

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

```
71 \newcommand{\num@bs}{%
72 \ifundefined{chapter}{\section{\abstractname}}{%
73 \chapter{\abstractname}}
74 }
75
```

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.

```
76 \if@titlepage
```

For the titlepage option.

```
77 \renewenvironment{abstract}{%
78   \titlepage
79   \null\vfil
80   \@beginparpenalty\@lowpenalty
```

Number the title, if called for, otherwise typeset the title in the specified (via `\absnamepos`) environment and adjust the following vertical spacing.

```
81   \ifnumber@bs \num@bs \else
82     \begin{\absnamepos}%
83       \abstractnamefont \abstractname
84       \@endparpenalty\@M
85     \end\absnamepos%
86     \vspace{\abstitleskip}%
87   \fi
```

Try adding the title to the ToC, then start the environment for typesetting the text.

```
88   \put@bsintoc%
89   \begin{@bstr@ctlist}\abstracttextfont}%
```

Finish the **abstract** environment.

```
90   {\par\end{@bstr@ctlist}\vfil\null\endtitlepage}
91 \else
```

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

```
92 \renewenvironment{abstract}{%
93   \ifnumber@bs \num@bs \else
94     \begin{\absnamepos}\abstractnamefont\abstractname\end\absnamepos%
95     \vspace{\abstitleskip}%
96   \fi
97   \put@bsintoc%
98   \begin{@bstr@ctlist}\abstracttextfont}%
99   {\par\end{@bstr@ctlist}}
100 \fi
101
```

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

```
102 \newenvironment{onocolabstract}{%
103   \begin{@twocolumnfalse}\begin{abstract}}{%
104   \end{abstract}\end{@twocolumnfalse}}
105
```

`\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.²

²Code suggestions for this were made by Michael Downes (epsmj@ams.org) and Heiko Oberdiek (oberdiek@ruf.uni-freiburg.de) on `ctt`.

```

106 \providecommand{\appendiargdef}[2]{\begingroup
107   \toks@\expandafter{#1{##1}#2}%
108   \edef\@bsx{\endgroup \def\noexpand#1####1{\the\toks@}}%
109   \@bsx}
110

```

`\thanks` We have to keep the contents of the `\thanks` commands as normally these are
`\@bs@thanks` 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`.

```

111 \appendiargdef{\thanks}{%
112   \protected@xdef\@bs@thanks{\@bs@thanks
113     \protect\footnotetext[\the\c@footnote]{#1}}%
114 }
115 \let\@bs@thanks\@empty
116

```

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

```

117 \newcommand{\saythanks}{\begingroup
118   \renewcommand{\thefootnote}{\fnsymbol{footnote}}\@bs@thanks
119   \endgroup\global\let\@bs@thanks\@empty}
120

```

The end of this package.

```

121 </usc>

```

References

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