

artthreads: Support for article threads

D. P. Story
Email: dpstory@acrotex.net

processed July 13, 2020

Contents

1	Introduction	1
2	Options and Required Packages	1
3	The Main Code	2
3.1	<code>\setThreadInfo</code> and its KV-pairs	3
3.2	The <code>\bArticle</code> and <code>\cArticle</code> commands	3
3.3	The driver dependent commands	5
3.4	Some navigation commands	7
4	Index	10
5	Change History	12
1	<code>*package</code>	

1 Introduction

This package is used to create *article threads* around blocks of content. Generally, article threads are used with multi-column documents, such as newspapers or newsletters, or anything that uses a multi-column format.

The creation of the article threads should occur as the final step of the document composition. There is nothing automatic about placing the threads, \TeX does not lend itself to the creation of the threads. For the purpose of placing the threads, the options `preview` and `viewMagWin` are integral.

2 Options and Required Packages

`2 \RequirePackage{xkeyval}`

Driver options. These options are `dvips` (the default), `pdftex`, `luatex`, and `xetex` are automatically detected, other drivers supported are `dvipdfm` and `dvipdfmx`.

Preview options. The `preview` option is a carry over from `eforms`. When selected, all form fields are outlined; useful when setting the location of fields in a dvi previewer. The other option is `viewMagWin` will show the viewing windows surrounding the target. This is the rectangle that will be jumped to. Use this option to adjust the size of the window to your needs. When either of the last two options has an exclamation point prior, that means to turn off the switch. You can conveniently use `viewMagWin` to see the viewing window, then change it to `!viewMagWin` to remove the visible window. Cool. Similarly, you can turn off `preview` using `!preview`.

```

pdftex All options are passed to the fitr package, which contains a lot of the code used in
luatex this package.
    3 \DeclareOptionX{pdftex}{\PassOptionsToPackage{\CurrentOption}{fitr}}
    4 \DeclareOptionX{luatex}{\PassOptionsToPackage{\CurrentOption}{fitr}}

dvips Distiller based drivers.
dvipsone 5 \DeclareOptionX{dvips}{\PassOptionsToPackage{\CurrentOption}{fitr}}
        6 \DeclareOptionX{dvipsone}{\PassOptionsToPackage{\CurrentOption}{fitr}}

xetex xetex and its variants.
dvipdfm 7 \DeclareOptionX{dvipdfm}{\PassOptionsToPackage{\CurrentOption}{fitr}}
dvipdfmx 8 \DeclareOptionX{dvipdfmx}{\PassOptionsToPackage{\CurrentOption}{fitr}}
        9 \DeclareOptionX{xetex}{\PassOptionsToPackage{\CurrentOption}{fitr}}

preview Preview preferences
!preview 10 \DeclareOptionX{preview}
viewMagWin 11 {\PassOptionsToPackage{\CurrentOption}{fitr}}
!viewMagWin 12 \DeclareOptionX{!preview}
            13 {\PassOptionsToPackage{\CurrentOption}{fitr}}
            14 \DeclareOptionX{viewMagWin}
            15 {\PassOptionsToPackage{\CurrentOption}{fitr}}
            16 \DeclareOptionX{!viewMagWin}
            17 {\PassOptionsToPackage{\CurrentOption}{fitr}}

Any other options are passed on to fitr. One useful option is gonative, which creates no document JavaScript. Consequently, the latex -> dvips -> ps2pdf workflow is supported.
18 \DeclareOptionX*{\PassOptionsToPackage{\CurrentOption}{fitr}}
19 \ProcessOptionsX

```

3 The Main Code

We require the package `fitr`, as we use many commands from that package. We pass all options to `fitr`.

```

20 \RequirePackage{fitr}[2020/07/09]
Some scratch counters, lengths, boxes.
21 \newif\ifnewarticle
22 \newcounter{@rtBtnCnt}

```

```

23 \def\CntArt{0}\def\CntArtInfo{0}
24 \def\inc@CntMacro#1{{\@tempcnta#1\relax\advance\@tempcnta1\relax
25   \xdef#1{\the\@tempcnta}}}
26 \def\artM@rk{[\space}

```

3.1 \setThreadInfo and its KV-pairs

`title` We set up key-values for the thread Info dictionary. Only the title is required, that's why a default value of `thread-⟨num⟩` is otherwise provided. For `pdflatex` and `lualtex`, only `title` is supported.

```

keywords 27 \define@key{artthrd}{title}[]{\Hy@unicodedefalse
28   \pdfstringdef\threadTitle{#1}}
29   \let\threadTitle\@empty
30 \define@key{artthrd}{author}[]{\pdfstringdef\threadAuthor{#1}}
31   \let\threadAuthor\@empty
32 \define@key{artthrd}{keywords}[]{\pdfstringdef\threadKeywords{#1}}
33   \let\threadKeywords\@empty
34 \define@key{artthrd}{subject}[]{\pdfstringdef\threadSubject{#1}}
35   \let\threadSubject\@empty

```

`\setThreadInfo` Use `\setThreadInfo` prior to the opening of a new thread using `\cArticle`. Through its key-values, set the title, author, keywords, and subject of the thread. For `pdftex`, only `title` is supported.

```

36 \newcommand{\setThreadInfo}[1]{\inc@CntMacro\CntArtInfo
37   \let\threadTitle\@empty\let\threadAuthor\@empty
38   \let\threadKeywords\@empty\let\threadSubject\@empty
39   \setkeys{artthrd}{#1}\ifx\threadTitle\@empty
40     \def\threadTitle{thread-\CntArtInfo}\fi
41 \def\bArtErrMsg{When starting a new article thread
42   (\string\bArticle)\MessageBreak
43   first use \string\setThreadInfo\space to set the title,\MessageBreak
44   author, subject, and keywords}

```

`addtow` **Additional key-values for threads** In addition to the key-values provided by `fitr`, we provide a few more. The two keys `addtow` and `addtoh` adds to the reading border of the threads. These are set through the command `\setAddToBorder`.

`addtoh`

`\setAddToBorder` For the most part, choices for `addtow` (add to the width) and `addtoh` (add to the height) should be set for the whole document. Initially, `addtow=0,addtoh=0`.

```

45 \define@key{artthrd}{addtow}[0]{%
46   \fitr@defaultbp{\artthrd@addtow}{#1}}
47 \def\artthrd@addtow{0}
48 \define@key{artthrd}{addtoh}[0]{%
49   \fitr@defaultbp{\artthrd@addtoh}{#1}}
50 \def\artthrd@addtoh{0}
51 \newcommand{\setAddToBorder}[1]{\setkeys{artthrd}{#1}}

```

3.2 The \bArticle and \cArticle commands

`\bArticle` The `\bArticle` is used to start an article thread. The thread may continue with

`\cArticle`. Both commands have a required argument, consisting of key-values defined by `fitr`. The following is the syntax, with appropriate keys defined in `fitr`.

```
\cArticle{lift=<length>,width=<length>,height=<length>}
```

```
52 \newcommand{\bArticle}{\begingroup\inc@CntMacro\CntArt
53   \ifnum\CntArt=\CntArtInfo\relax\else
54   \PackageError{artthreads}{\bArtErrMsg}{See package documentation
55   for \string\setThreadInfo}\fi
56   \newarticlettrue\set@rectart}
```

`\cArticle` The only difference between `\cArticle` and `\bArticle` is the stepping of the counter `\CntArt` and the setting of the switch `\fnewarticle`.

```
57 \newcommand{\cArticle}{\begingroup\newarticlefalse\set@rectart}
```

`\set@rectart` continues `\bArticle`/`\cArticle`.

```
58 \newcommand{\set@rectart}[1]{\def\fitr@artrect@argi{#1}% dps
59   \edef\temp@exp{\noexpand\setkeys{fitr}{#1}}\temp@exp
60   \set@rectart}
```

`\set@@rectart` is the main command for building the thread.

```
61 \def\set@@rectart{\def\fitr@setBL{Opt}%
62   \@ifundefined{ef@Bbox}{\let\fitr@FLB@ction\Bbox}%
63   {\let\fitr@FLB@ction\ef@Bbox}%
64   \get@fitr@dimen{%
65     \parbox[\fitr@refPt][\fitr@height]{\fitr@@width}%
66     {\kernOpt\fitr@FLB@ction{\fitr@@width}{\fitr@height}\kernOpt}%
67   }%
68   \makebox[Opt][l]{\hspace*{\fitr@@shift}\smash{%
69     \raisebox{\fitr@@lift-\fitr@setBL}{%}
```

Actually insert the thread code here by inserting `\setThre@d`.

```
70   \setThre@d{\artthrd@@addtow}{\artthrd@@addtoh}%
71   \unhbox\fitr@bbox
72   \fitr@urxury@fixup{\artthrd@@addtow}{\artthrd@@addtoh}{}%
73   }}\setlength{\fboxsep}{Opt}%
```

Now place an extended border around the viewing area

```
74 \ifviewMagWin
75   \fitr@length=\artthrd@@addtow bp
76   \fitr@length=2\fitr@length
77   \addtolength{\fitr@length}{\fitr@@width}%
78   \edef\fitr@@width{\the\fitr@length}%
79   \fitr@length=\artthrd@@addtoh bp
80   \fitr@length=2\fitr@length
81   \addtolength{\fitr@length}{\fitr@@height}%
82   \edef\fitr@@height{\the\fitr@length}%
83   \setlength\fitr@length{-\artthrd@@addtow bp}%
84   \addtolength\fitr@length{\fitr@@shift}%
85   \makebox[Opt][l]{\hspace*{\fitr@length}\smash{%
86     \setlength\fitr@length{\fitr@@lift-\fitr@setBL}%
87     \if\fitr@@refPt b%
```

```

88     \addtolength\fitr@length{-\artthrd@addtoh bp}\else
89     \if\fitr@refPt t%
90     \addtolength\fitr@length{\artthrd@addtoh bp}%
91     \fi\fi
92     \raisebox{\fitr@length}%
93     {\fbox{\parbox[\fitr@refPt][\fitr@height]%
94     {\fitr@width}{\kern0pt\hfill\vfill\kern0pt}}}%
95     }%
96     \fi
97 \endgroup}

```

3.3 The driver dependent commands

We begin with a required definition to enable lualatex to function correctly.

```
98 \ifluatex\protected\def\pdfthread{\pdfextension thread }\fi
```

`\AT@ThrBbox{<add-wd>{<add-ht>}}` The definition of `\set@rtThrd` depends on the driver. The arguments are dimensionless numbers, representing Adobe points. The `\AT@ThrBbox` macros perform different tasks, one of which is to calculate the dimensions of the bounding box for the viewing rectangle. It is used in `\set@rtThrd`, defined below.

```
99 \if@fitr@dvi pdfm
```

For the xelatex and friends

```

100 \def\AT@ThrBbox#1#2{%
101   \fitr@length=#1bp
102   \fitr@length=2\fitr@length
103   \advance\fitr@length\wd\fitr@bbox
104   \edef\fitr@xetex@view@width{\the\fitr@length}%
105   \fitr@length=#2bp
106   \advance\fitr@length\ht\fitr@bbox
107   \edef\fitr@xetex@view@height{\the\fitr@length}%
108   \fitr@length=#2bp
109   \advance\fitr@length\dp\fitr@bbox
110   \edef\fitr@xetex@view@depth{\the\fitr@length}%
111 } \else
112 \ifpdf

```

For the pdftex and friends

```

113 \def\AT@ThrBbox#1#2{%
114   \fitr@length=#1bp
115   \fitr@length=2\fitr@length
116   \advance\fitr@length\wd\fitr@bbox
117   \edef\fitr@pdftex@view@width{\the\fitr@length}%
118   \fitr@length=#2bp
119   \advance\fitr@length\ht\fitr@bbox
120   \edef\fitr@pdftex@view@height{\the\fitr@length}%
121   \fitr@length=#2bp
122   \advance\fitr@length\dp\fitr@bbox
123   \edef\fitr@pdftex@view@depth{\the\fitr@length}%

```

```

124 }\else
125 \ifx\fitr@driver\fitr@dviptionsone@driver
    For the dviptionsone driver
126 \def\AT@ThrBbox#1#2{%
127   currentpoint 2 copy %
128   \the\fitr@depth\space add #2\space PDFtoTeX add      % y1
129   exch #1\space PDFtoTeX sub exch                      % x1
130   4 2 roll exch
131   \the\fitr@width\space add #1\space PDFtoTeX add exch % x2
132   \the\fitr@height\space sub #2\space PDFtoTeX sub}    % y2
133 \else
134 \ifx\fitr@driver\fitr@dviptions@driver
    For the dviptions driver
135 \def\AT@ThrBbox#1#2{% Uses \fitr@bbox
136   currentpoint 2 copy %
137   \the\fitr@depth\space SPToDviptions add
138     #2\space BPToDviptions add                        % y1
139   exch #1\space BPToDviptions sub exch                % x1
140   4 2 roll exch
141   \the\fitr@width\space SPToDviptions add
142     #1\space BPToDviptions add exch                  % x2
143   \the\fitr@height \space SPToDviptions sub
144     #2\space BPToDviptions sub}                      % y2
145 \fi\fi\fi\fi

```

Code to set the thread The `\setThread` macros are driver dependent.

`\setThread{<add-wd>{<add-ht>}}` This macro strips out the dimensions of its arguments (10bp becomes 10) and passes these new arguments to `\set@rtThrd`.

```

146 \def\setThread#1#2{%
147   \fitr@defaultbp{\aWd}{#1}\fitr@defaultbp{\aHt}{#2}%
148   \edef\tmp@exp{\noexpand\set@rtThrd{\aWd}{\aHt}}\tmp@exp}
149 \if@fitr@dviptions

```

`\set@rtThrd{<add-wd>{<add-ht>}}` The definition of `\set@rtThrd` depends on the driver. The arguments are dimensionless numbers, representing Adobe points.

For the xelatex and friends

```

150 \def\set@rtThrd#1#2{\AT@ThrBbox{#1}{#2}%
151   \smash{\raisebox{-\fitr@xetex@view@height+#2bp}{%
152     {\makebox[0pt][l]{\hspace*{#1bp}}%
153     \@pdfm@mark{thread @artthrd\CntArt\space
154     width \fitr@xetex@view@width\space
155     height \fitr@xetex@view@height\space
156     depth \fitr@xetex@view@depth\space
157     << /Title (\threadTitle)
158     \ifnewarticle
159     /Author (\threadAuthor)
160     /Subject (\threadSubject)
161     /Keywords (\threadKeywords)\fi >>}}}}}\else

```

```

162 \ifpdf
    For pdftex and friends
163 \def\set@rtThrd#1#2{\AT@ThrBbox{#1}{#2}%
164 \makebox[0pt][l]{\hspace*{-#1bp}%
165 \pdfthread
166 width \fitr@pdftex@view@width\space
167 height \fitr@pdftex@view@height\space
168 depth \fitr@pdftex@view@depth\space
169 name {\threadTitle}}\else
    For drivers that use PostScript and the pdfmark.
170 \def\set@rtThrd#1#2{\literalps@out{\artM@rk
171 /Rect [\AT@ThrBbox{\artthrd@addt@w}{\artthrd@addt@h}]
172 /Title (\threadTitle)
173 \ifnewarticle
174 /Author (\threadAuthor)
175 /Subject (\threadSubject)
176 /Keywords (\threadKeywords)\fi
177 /ARTICLE pdfmark}}
178 \fi\fi

```

3.4 Some navigation commands

```
179 \let\threadTitle\@empty
```

`\Thread{<thread-title>}` The action to start read a thread whose title is `<thread-title>`.
For example, `\setLink[\A{\Thread{Lipsum}}]{Read Lipsum}`

```
180 \providecommand{\Thread}[1]{/S/Thread/D(#1)}
```

```

\shArticlesPaneActn Actions for show/hid articles pane and to just show articles pane (The named
\sArticlesPaneActn action ArticleThreads is undocumented, don't remember show I got this name.)
\shArticlesPaneReadActn In addition to these two, we combine them with read thread actions:
\sArticlesPaneReadActn 181 \def\shArticlesPaneActn{\Named{ShowHideArticles}}
182 \def\sArticlesPaneActn{\Named{ArticleThreads}}
183 \def\shArticlesPaneReadActn{\shArticlesPaneActn
184 \Next{\Thread{\threadTitle}}}
185 \def\sArticlesPaneReadActn{\sArticlesPaneActn
186 \Next{\Thread{\threadTitle}}}

```

`\readArticle{<thread-title>}` is a macro intended to be passed to a push button action through the `\cmd` key. For example,

```
\showArticlePaneRead[\cmd{\readArticle{more Lipsum}}]{11bp}
```

It should be used with `\toggleArticlePaneRead` and `\showArticlePaneRead` (push buttons). Alternatively, these actions can be done through a link:

```

\setLink[\cmd{\readArticle{Lipsum}}
\A{\shArticlesPaneReadActn}]{Lipsum}

```

```

187 \def\readArticle#1{\Hy@unicodedefalse
188 \pdfstringdef\AT@x{#1}\edef\threadTitle{\AT@x}}
189 \def\chkThreadName{%
190 \ifx\threadTitle\empty
191 \global\let\isThrTtl\ef@NO
192 \def\readArtPresets{\cmd{\let\Next@gobble}}\else
193 \global\let\isThrTtl\ef@YES
194 \let\readArtPresets\empty
195 \fi
196 }
197 \def\readThreadMsg{\ifx\isThrTtl\ef@NO
198 \PackageWarning{artthreads}{No thread title specified to read}\fi}

\toggleArticlePane[⟨KV-pairs⟩]{⟨wd⟩}{⟨ht⟩} A push button whose action toggles the Articles pane
open and closed.
199 \newcommand{\toggleArticlePane}[3][\stepcounter{@rtBtnCnt}]%
200 \pushButton[\CA{Article}\TU{Toggle Articles Pane}#1
201 \A{\shArticlesPaneActn}
202 ]{TAPBtn\the@rtBtnCnt}{#2}{#3}}

\toggleArticlePaneRead[⟨KV-pairs⟩]{⟨wd⟩}{⟨ht⟩} A push button whose action us to toggle the Articles
pane state (show/hide) and to begin reading the thread titled \threadTitle. If
this command is placed after the \setThreadInfo command, then \threadTitle
is the title set by the title key of \setThreadInfo.
203 \def\tooltipTogglePaneRead#1{\def\ttToggleP@neRead{#1}}
204 \tooltipTogglePaneRead{Toggle Articles pane and read
205 the article \threadTitle}
206 \newcommand{\toggleArticlePaneRead}[3][\begingroup
207 \stepcounter{@rtBtnCnt}\pushButton[\CA{Article}#1
208 \TU{\ttToggleP@neRead}\cmd{\chkThreadName}
209 \presets{\readArtPresets}\A{\shArticlesPaneReadActn}
210 ]{TAPRBtn\the@rtBtnCnt}{#2}{#3}\readThreadMsg\endgroup}

\showArticlePane[⟨KV-pairs⟩]{⟨wd⟩}{⟨ht⟩} A push button whose action is to show the Articles pane.
211 \newcommand{\showArticlePane}[3][\stepcounter{@rtBtnCnt}]%
212 \pushButton[\CA{Article}\TU{Show Articles Pane}#1
213 \A{\sArticlesPaneActn}
214 ]{SAPBtn\the@rtBtnCnt}{#2}{#3}}

\showArticlePaneRead[⟨KV-pairs⟩]{⟨wd⟩}{⟨ht⟩} A push button whose action is to show the Articles and
to begin reading the thread titled \threadTitle. If this command is placed after
the \setThreadInfo command, then \threadTitle is the title set by the title key
of \setThreadInfo.
215 \def\tooltipShowPaneRead#1{\def\ttShowP@neRead{#1}}
216 \tooltipShowPaneRead{Show Articles pane and read
217 the article \threadTitle}
218 \newcommand{\showArticlePaneRead}[3][\begingroup
219 \stepcounter{@rtBtnCnt}\pushButton[\CA{Article}#1
220 \TU{\ttShowP@neRead}\cmd{\chkThreadName}
221 \presets{\readArtPresets}\A{\sArticlesPaneReadActn}
222 ]{SAPRBtn\the@rtBtnCnt}{#2}{#3}\readThreadMsg\endgroup}

```


223 \langle /package \rangle

4 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.

Symbols	
<code>\@pdfm@mark</code>	153
<code>!preview</code> (option)	<i>2</i>
<code>!viewMagWin</code> (option)	<i>2</i>
A	
<code>\A</code>	201, 209, 213, 221
<code>addtoh</code> (key)	<i>3</i>
<code>addtow</code> (key)	<i>3</i>
<code>\aHt</code>	147, 148
<code>\artM@rk</code>	26, 170
<code>\artthrd@@addtoh</code>	49, 50, 70, 72, 79, 88, 90, 171
<code>\artthrd@@addtow</code>	46, 47, 70, 72, 75, 83, 171
<code>\AT@ThrBbox</code>	5, 100, 113, 126, 135, 150, 163, 171
<code>\AT@x</code>	188
<code>author</code> (key)	<i>3</i>
<code>\aWd</code>	147, 148
B	
<code>\bArtErrMsg</code>	41, 54
<code>\bArticle</code>	<i>3</i> , 42, 52
<code>\Bbox</code>	62
C	
<code>\CA</code>	200, 207, 212, 219
<code>\cArticle</code>	<i>4</i> , 57
<code>\chkThreadName</code>	189, 208, 220
<code>\cmd</code>	192, 208, 220
<code>\CntArt</code>	23, 52, 53, 153
<code>\CntArtInfo</code>	23, 36, 40, 53
<code>\CurrentOption</code>	3–9, 11, 13, 15, 17, 18
D	
<code>\DeclareOptionX</code>	3–10, 12, 14, 16, 18
<code>dvipdfm</code> (option)	<i>2</i>
<code>dvipdfmx</code> (option)	<i>2</i>
<code>dvips</code> (option)	<i>2</i>
<code>dvipsone</code> (option)	<i>2</i>
E	
<code>\ef@Bbox</code>	63
<code>\ef@NO</code>	191, 197
<code>\ef@YES</code>	193
F	
<code>\fbox</code>	93
<code>\fboxsep</code>	73
<code>\fitr@height</code>	65, 66, 81, 82, 93
<code>\fitr@lft</code>	69, 86
<code>\fitr@refPt</code>	65, 87, 89, 93
<code>\fitr@shift</code>	68, 84
<code>\fitr@width</code>	65, 66, 77, 78, 94
<code>\fitr@artrect@argi</code>	58
<code>\fitr@bbox</code>	71, 103, 106, 109, 116, 119, 122, 135
<code>\fitr@defaultbp</code>	46, 49, 147
<code>\fitr@depth</code>	128, 137
<code>\fitr@driver</code>	125, 134
<code>\fitr@dvi@sdriver</code>	134
<code>\fitr@dvi@sone@driver</code>	125
<code>\fitr@FLB@ction</code>	62, 63, 66
<code>\fitr@height</code>	132, 143
<code>\fitr@length</code>	75–86, 88, 90, 92, 101–110, 114–123
<code>\fitr@pdftex@view@depth</code>	123, 168
<code>\fitr@pdftex@view@height</code>	120, 167
<code>\fitr@pdftex@view@width</code>	117, 166
<code>\fitr@setBL</code>	61, 69, 86
<code>\fitr@curxury@fixup</code>	72
<code>\fitr@width</code>	131, 141
<code>\fitr@xetex@view@depth</code>	110, 156
<code>\fitr@xetex@view@height</code>	107, 151, 155
<code>\fitr@xetex@view@width</code>	104, 154
G	
<code>\get@fitr@dimen</code>	64
H	
<code>\Hy@unicodedefalse</code>	27, 187
I	
<code>\if@fitr@dvi@pdfm</code>	99, 149
<code>\ifluatex</code>	98
<code>\ifnewarticle</code>	21, 158, 173
<code>\ifpdf</code>	112, 162
<code>\ifviewMagWin</code>	74
<code>\inc@CntMacro</code>	24, 36, 52
<code>\isThrTtl</code>	191, 193, 197
K	
<code>\kern</code>	66, 94
keys:	
<code>addtoh</code>	<i>3</i>

5 Change History

v1.1 (2016/04/09)		v1.3 (2020/07/03)	
General: Removed the Page key and dependence on the <code>recount</code> package	7	General: Add special definition for threads in the case of <code>lualatex</code>	5
v1.2 (2016/05/02)		Added <code>\Thread</code>	5
General: Modified documentation to reflect new DC thread icons.	1	Use <code>\ifpdf</code>	5, 7
		v1.3.1 (2020/07/09)	
		General: Require <code>fitr</code> package, dated 2020/07/09	2