

This work (i.e., all the files in the `xwatermark` manifest) may be distributed and/or modified under the conditions of the L^AT_EX Project Public License (LPPL), either version 1.3 of this license or any later version. The LPPL maintenance status of this software is ‘author-maintained’. This software is provided ‘as it is,’ without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

The `xwatermark` Package © ☆

Version 1.4

Ahmed Musa

University of Central Lancashire
Preston PR1 2HE, United Kingdom

a.musa@rocketmail.com

January 15, 2010

Abstract

The `xwatermark` package enables the user to effortlessly put customized text and graphic watermarks on select pages of documents at desired positions and orientations. The features of the individual watermarks are specified by means of user-friendly key-value interfaces.

Contents

1	Introduction	2
2	User interface	3
2.1	Loading the package	3
2.2	Graphic watermarks	5
2.3	The <code>\watermarksetup</code> macro	6
2.4	Options without values	7
2.5	Emptying the watermarks of some pages	7
2.6	The usefulness of the <code>white</code> color	8
3	Other aspects of package architecture and use	8
3.1	<code>\documentclass</code> options	8
3.2	The size of the watermark	8
3.3	The coordinates of the watermark	9
3.3.1	Wrong location of the watermark	9
3.4	Choice of <code>grayness</code>	9

☆The `xwatermark` bundle is available at the Comprehensive T_EX Archive Network
<<http://tug.ctan.org/tex-archive/macros/latex/contrib/xwatermark>>

3.5	Breaking the watermark into lines	10
3.5.1	The alignment of the watermark	10
3.6	Locating the page center	10
3.7	Active characters	11
4	The <code>\xwcolorbox</code> and <code>\xwminipage</code> macros	12
4.1	The <code>\ywcolorbox</code> macro	14
5	Using <code>xwatermark</code> with the <code>geometry</code> package	15
6	Support for Unicode and UTF encodings	16
7	Further examples of use of <code>xwatermark</code> package	17
8	Future extension	17
9	Epilogue	17
10	Acknowledgement	18
A	Package options	18
A.1	Global options	18
A.2	Local options	21

1 Introduction

THIS PACKAGE puts user-specified watermarks, pictures and/or arbitrary texts on select pages of documents. It has more functionality and dynamism than, for example, the packages `draftcopy`, `draftwatermark`, `watermark`, and `draftmark`¹. The advantages of `xwatermark` package over these earlier packages include: both text and graphic watermarks are admissible; the user can specify his/her own watermark and color, the watermark position, orientation, the page(s) (all pages, odd pages, even pages, a particular page, and a range of pages) on which the watermark should appear; and all the options are passed directly to the package instead of being defined in the source file by several macros. Moreover, all these options can be customized for individual pages and segments of the document. This is achieved by the provision of the user-friendly `\watermarksetup` macro, by which the user can dynamically set the watermark properties for each page or range of pages. The user is relieved of the need to remember and deploy several macros.

With the `xcolor` package (loaded by this package), all colors (including `white`, shades like `-red!75!green!50`, and those defined within the user document) can be passed to this package. And, as mentioned above, both texts and pictures can be submitted as watermarks to the package.

¹These and other packages are available at the TUG CTAN website (<http://tug.ctan.org/cgi-bin/>).

This package uses Heiko Oberdiek's shipout hooks from the `atbegshi` package.

There are global and local package options. These are listed and explained in Appendix A.

2 User interface

THE SUBSECTIONS under this section describe how the user may interface with (i.e., use) the facilities of the package.

2.1 Loading the package

In the case of text watermarks, the package may be loaded with options such as in

```

1 \usepackage[
2   printwatermark=true,allpages=true,fontfamily=pag,
3   color=gray!25,textmark=I am happy,angle=45,fontsize=5cm,
4   width=\paperwidth,fontseries=b,scale=0.8,
5   xcoord=0,ycoord=0
6 ]{xwatermark}

```

or

```

7 \usepackage[printwatermark,pages=1-4,fontfamily=phv,
8   fontsize=12pt,width=\paperheight,angle=55,scale=0.8,
9   xcoord=20,ycoord=10,color=red!55!yellow!50,
10  textmark=CONFIDENTIAL]{xwatermark}.

```

By design, the boolean option `printwatermark` should not appear in the macro `\watermarksetup` but in the statement `\usepackage{xwatermark}` or in the `\documentclass` options list. It is disabled before `\begin{document}` and any attempt to change it thereafter will trigger an error.

When boolean options (e.g., `printwatermark` and `allpages`) are passed without values, they are assumed implicitly `true` by the package. The option `printwatermark` may be set (`printwatermark` or `printwatermark=false`) when invoking the package, within the options list of either `\documentclass` or `\usepackage{xwatermark}`. Otherwise, the package will assume it to be `true`.

The `textmark` key applies to text watermarks, for which all the font properties can be selected. It does not apply to graphic watermarks. For graphic watermarks you need the keys: `picfile` (the graphic/picture filename, with

its full path but without its extension), and `picfilex` (the file extension). Admissible file extensions are `eps`, `pdf`, `png`, and `jpeg`; they should be submitted without the dot. Additional information is needed (see section 2.2).

The following points should be noted about the values of the `textmark` key:

- The value of the `textmark` may be any arbitrary multiline text, such as

```
11 \textmark={Hello world,\\[.25\baselineskip] We're here},
```

- The value of `textmark` may be arbitrary (blocks of) texts or even kernel or package commands, but not filenames on their own (except when submitted within, e.g., `\includegraphics`).
- If the `textmark`'s value contains one or more commas, then it has to be enclosed in braces (as above), otherwise (L^A)T_EX will report an error. In any case, it is always safer to enclose long mark texts in braces. However, *active characters* need further special treatment (see section 3.7).
- The `width` (=markwidth) should be properly selected to match user's taste and the length of the `textmark`. It may be set to `\paperwidth` or `\paperheight`, or any arbitrary length. Its default value is preset to `\paperheight`. Sometimes it might also be necessary to suitably select the `height` (=markheight), whose default value is `\paperwidth`.
- If the longest line of a `textmark` is longer than `\paperwidth` and/or `\paperheight` (depending on the orientation of the `textmark`), then the `fontsize` and `scale` options will have to be suitably chosen.

The package boolean option `allpages` in

```
12 \usepackage[printwatermark,allpages]{xwatermark},
```

which specifies the pages that should receive watermarks, may be replaced by any of the following options:

```
13 \firstpage, oddpages, evenpages, page=x, pages=x-x,
```

where `x` means any page number. If you enter, for example, `pages=0-10`, all pages from 1 to 10 will have watermark. On the other hand, an entry like `pages=10-0` will print watermark on page 10 only. If no page-specifying option is given, watermark will appear only on the first page and a warning message will be entered in the transcript file. When passing `page=x` or `pages=x-x` as option to package, don't forget to include the equality sign (=), otherwise the option will be ignored by the package and a warning message will be logged in the transcript file.

When setting package options either in `\usepackage` or `\documentclass` (or indeed in `\watermarksetup`, `\xwminipage` or `\xwcolorbox`), the following points should be noted:

- Multiple lines are permitted but not blank lines.
- Extra spaces between options and words are ignored.
- No active characters (those of catcode 13) are allowed (but see section 3.7).
- Options are basically order-agnostic.

Thus, e.g., the following statement

```

14 \usepackage
15   [ printwatermark = true, scale = 0.8,
16   xcoord = 0, ycoord = 0, fontfamily = pag,
17     allpages = true,
18   color = gray!25, mark = I am happy, angle = 45,
19     fontsize=5cm,
20     width = \paperwidth, fontseries = b]
21 {xwatermark}

```

is equivalent to

```

22 \usepackage[printwatermark,allpages=true,fontfamily=pag,
23   color=gray!35,textmark=I am happy,angle=45,fontsize=5cm,
24   width=\paperwidth,fontseries=b,scale=0.8,
25   xcoord=0,ycoord=0]{xwatermark}.

```

2.2 *Graphic watermarks*

For graphic/picture watermarks, the user can issue, e.g.,

```

26 \usepackage[
27   printwatermark,page=2,
28   picfile={.../graphicsdir/pic-file},picfilex=eps,
29   picbb=116 619 242 751,picscale=3,align=center,angle=0,
30   xcoord=0,ycoord=0
31 ]{xwatermark}.

```

For graphic and picture watermarks, you need the `picfile` (the graphic filename, with its full path but without its extension), `picfilex` (the picture filename extension without the dot), `picbb` (the picture bounding box), and `picscale` (the picture scale). Admissible file extensions are `eps`, `pdf`, `png` and

`jpeg`; the latter three, but not the first, may be used in the case of `pdfTeX`. The file extension should be passed without the dot. If the file extension is not passed to package, the package selects it automatically based on whether `pdfTeX` mode running or not (normal extensions are `eps` for `dvi` mode and `pdf` for `pdfTeX` mode). If you have the graphic file in both `eps` and `pdf` formats, then you don't have to bother about submitting the file extension to the package: it will automatically select the appropriate file extension, depending on the mode (`pdfTeX` or `dvi`) in which it is running.

2.3 The `\watermarksetup` macro

The global boolean option `printwatermark=true` (or `=false`) and the one that specifies the pages to receive the watermark(s) (e.g., `evenpages` and `allpages`) should be set when loading the package, e.g.,

```
32 \usepackage[printwatermark,evenpages]{xwatermark},
```

or in the `\documentclass` options list:

```
33 \documentclass[a4paper,12pt,printwatermark]{article}
```

```
35 \usepackage[evenpages]{xwatermark}.
```

The remaining options can be set dynamically using the `\watermarksetup` macro. These other options can be set for each page, as on the pages of the accompanying file of examples (`xwatermark-examples.tex`). The macro `\watermarksetup` can be used as in

```
36 \watermarksetup{fontfamily=bch,color=gray!25,textmark=DRAFT,
37   angle=45,scale=0.8,xcoord=0,ycoord=0
38 }.
```

The option `printwatermark` should not appear in `\watermarksetup` but in the statement `\usepackage{xwatermark}` or in the `\documentclass` options list.

However, the options `allpages`, `oddpages`, `evenpages`, etc., which specify watermark pages, can appear in `\watermarksetup`. This implies that the instructions that specify watermark pages may be issued and superseded dynamically (page by page or chapter by chapter). For small documents, this feature may be unnecessary, but will be useful in large documents (such as a report or book), in which the watermark may change from chapter to chapter.

When you want the watermark on only one page of the document, you can conveniently use the `\watermarksetup` macro with the page option `page=desired page` in the preamble of your document after issuing

```
39 \usepackage[printwatermark]{xwatermark}.
```

In this way, you don't have to bother with locating in the source file the spot that corresponds to the page on which you want the watermark to appear.

2.4 *Options without values*

If you follow an option key with an equality sign but without a value, as in, e.g.,

```
40 \watermarksetup{fontfamily={},color=gray!25,textmark={},
41   angle=45,scale=0.8,xcoord=0,ycoord=0
42 },
```

then there will be no problem but the outcome may be unpredictable, depending on the key that has no value. In the above example, no watermark will be printed (not even the default mark, which is **DRAFT**) because the entry `textmark=` is valid and implies that no watermark should be printed. The absence of `fontfamily` in `fontfamily=` will compel (L^A)T_EX to use an arbitrary fontfamily that isn't the default (the default fontfamily is `phv` if the key `fontfamily` is not passed, and `cmr` otherwise).

2.5 *Emptying the watermarks of some pages*

If you issue any of the statements

```
43 allpages=true, evenpages=true, oddpages=true, pages=x-x,
44 page=x
```

together with `printwatermark=true` but you don't want the mark on any particular page, you can simply set `\watermarksetup{textmark=}` or, to the same effect, `\watermarksetup{textmark={}}` on that page. In the case of picture watermarks, you will set `\watermarksetup{picfile={}}`. These both imply that the watermark for that page is empty. This can be useful when transiting from text watermarks to picture watermarks or vice versa. Some examples are available in `xwatermark-examples.tex`, the example source file that shipped with this package.

2.6 *The usefulness of the white color*

If you set `allpages=true` or `evenpages=true` or `oddpages=true` together with `printwatermark=true` but you don't want the mark on any particular page, you can simply enter `color=white` in the `\watermarksetup` on that page. This applies only to text watermarks, as such a declaration has no effect on picture watermarks. This may be convenient in circumstances where you may change your mind as to whether to place a watermark on a particular page or not. In this way you don't have to set `\watermarksetup{textmark=}` or remove (or comment out) the `\watermarksetup` command for that (or indeed any) page.

3 Other aspects of package architecture and use

3.1 `\documentclass` options

The package is set to inherit the `\documentclass` options, if the options apply to the package. Therefore, some of the package options can be passed to the package via the `\documentclass` options list. This is perhaps most appropriate in the case of the options `printwatermark`. However, package options supersede those passed via the `\documentclass`. For example, the option `printwatermark=true` in the `\documentclass` options list can normally be superseded by the option `printwatermark=false` in loading the `xwatermark` package, e.g., as in

45 `\usepackage[printwatermark=false]{xwatermark},`

and vice versa.

If you don't need the watermark on any page of your document, simply replace the `printwatermark(=true)` option in `\usepackage` or `\documentclass` with the choice `printwatermark=false`. If you have specified `printwatermark(=true)` in the `\documentclass` options list but you still don't need the watermark on any page of your document, then you would have to enter `printwatermark=false` when loading the package.

3.2 *The size of the watermark*

In the case of text watermarks, the size of the watermark is controlled by three parameters, namely, `fontsize`, `fontseries` and `scale`. All can be set dynamically. Their default values are `5cm`, `b` and `1`, respectively. For picture watermarks, the size is determined by `picscale`.

3.3 *The coordinates of the watermark*

The watermark coordinates (specified by `xcoord` and `ycoord`) have their origin at the center of the page and are with respect to the geometric center of the watermark. The default unit is `millimeter`, but this can be changed on any page by issuing

```
46 \watermarksetup{coordunit=unit of length}.
```

Acceptable units of length include `mm` (millimeter), `cm` (centimeter), `in` (inch), `pt` (point), `bp` (big point), `dd` (didot), `ex` (height of small x), `pc` (pica), `cc` (cicero), `em` (width of capital M). The unit of coordinates can also be changed by simply submitting the unit to the package with other options at each call, as in

```
47 \watermarksetup{coordunit=pc,fontfamily=cmss,angle=90,
48   scale=1.0,textmark=-Official-,color=red!75!green!50,
49   xcoord=-10,ycoord=10
50 }.
```

3.3.1 *Wrong location of the watermark*

If you discover that the watermark is wrongly positioned on the page(s) of your document, the chances are that you have submitted wrong coordinates (values of `xcoord` and `ycoord`) to the package or the markwidth (`width`) is not optimal or both reasons. The package does not take responsibility for this and will normally not warn you in this respect. Since the output file provides a direct and simple indication of the occurrence of this error, no attempt has been made in the package to warn users in this regard. If you do not specify the keys `xcoord` and/or `ycoord` at all in the call to the package, their immediate past values will be used by the package. On the other hand, if you list these keys without their values in the call to the package, their default values (`xcoord=0` and `ycoord=0`, which yield the center of paper) will be assumed by the package. The default value of the markwidth is `\paperheight`.

When the `geometry` package is loaded together with `xwatermark`, page layout scale changes by the `geometry` package may result in the watermarks being positioned wrongly. See section 5 for further details.

3.4 *Choice of grayness*

The grayness applies to only text watermarks and is chosen according to the scheme of the `xcolor` package. The color gray is represented as `gray!x`, where

`x` is the grayness in percentage—normally from 1 to 100. In the previous versions of the package, the `grayness` was a package option associated with the color `gray`; this is no longer the case.

3.5 *Breaking the watermark into lines*

It is possible to break text watermarks into lines, as in the following examples:

```

51 \watermarksetup{fontfamily=ptm,angle=45,scale=.7,
52   textmark={Directorate\ [.25ex]Only},align=center,
53   color=green,xcoord=0,ycoord=0
54 }

56 \watermarksetup{fontfamily=ptm,angle=45,scale=.8,
57   textmark={Control\ [.25ex]Version},align=left,
58   color=green,xcoord=0,ycoord=0
59 }.

```

More complex examples are given in the file `xwatermark-examples.tex`.

3.5.1 *The alignment of the watermark*

The alignment of the watermark is controlled by the key `align`, which may be set to `center`, `left`, `right` or `justified`. The default is `center`. This is particularly useful for putting arbitrary texts (that are not necessarily watermarks) on pages of documents. Examples include the watermarks on the pages of this document. The code for the watermarks on page 1 is given in section 4.

3.6 *Locating the page center*

In case you need to locate the paper/page center for placing the watermark or some other material at any position on the page, a two-line grid can be placed on the page background with the key `showcenter`, which may be issued (dynamically for each page) with the `\watermarksetup` macro as follows:

```

60 \watermarksetup{showcenter} or

62 \watermarksetup{showcenter=true} or

64 \watermarksetup{showcenter,fontfamily=ptm,angle=60,
65   scale=.7,textmark=Confidential!,color=brown!25!yellow!75,

```

```
66   coordunit=cc,xcoord=0,ycoord=0
67   }.
```

An example is given in the example file (`xwatermark-examples.tex`). If after issuing this command to get a centered grid on a page, you no longer require the grid on the following pages, you simply issue another

```
68   \watermarksetup{showcenter=false} or
70   \watermarksetup{showcenter=false,fontfamily=panr,angle=60,
71     scale=.7,textmark=Confidential!,color=brown!25!yellow!75,
72     coordunit=cc,xcoord=0,ycoord=0
73   }.
```

3.7 *Active characters*

Active characters (i.e., those of category 13) can normally be used as values of the `textmark` key in the `\watermarksetup` macro. However, such values cannot be passed via the `\documentclass` or the `\usepackage{xwatermark}` statement. The `xkvltxp` package (or `kvoptions-patch` package) can be used to obviate this problem. That is, the following should work:

```
74   \RequirePackage{xkvltxp}
75   \documentclass[textmark={My watermark,\\designed~
76     by \textsc{Johnson}}]{class-type}
77   \begin{document}
78     ...
79   \end{document}
```

To pass active characters and expandable commands as values of `textmark` to the `xwatermark` package via `\usepackage{xwatermark}`, it is necessary to first load the `xkvltxp` package as above, or as follows:

```
80   \documentclass{class-type}
81   \usepackage{xkvltxp}
82   \usepackage[printwatermark,%
83     textmark=\textsc{Johnson's~Text}]{xwatermark}
84   \begin{document}
85     ...
86   \end{document}
```

The use of the `xkvltxp` package can be sidestepped completely by passing active characters not through `\documentclass` and `\usepackage{xwatermark}`

but instead via the `\watermarksetup` macro, as in

```

87 \documentclass{class-type}
88 \usepackage[printwatermark]{xwatermark}
89 \watermarksetup{allpages,textmark={M\"uller's~Text}}
90 \begin{document}
91 ...
92 \end{document}

```

In fact, in this way even expandable commands can be passed to the package, as follows:

```

93 \documentclass{class-type}
94 \usepackage[printwatermark]{xwatermark}
95 \newcommand*\temparg{Fran\c coise M\"uller's\endgraf Text}
96 \watermarksetup{allpages,textmark=\temparg}
97 \begin{document}
98 ...
99 \end{document}

```

The accompanying file `xwatermark-examples.tex` contains some complex examples.

In plain $\text{T}_{\text{E}}\text{X}$ the only active character is the tie character `~` (that is, `\nobreakspace`). However, some packages do make some other characters active. For example, after issuing the command `\MakeShortVerb{x}`, the packages `doc` and `shortvrb` make the character `x` active². The user can use such active characters in values of the `textmark` key without locally changing their catcode to 11 (i.e., letters). In the case of `\MakeShortVerb{x}`, you can issue `\DeleteShortVerb{x}` to revert to normal use of character `x`. As another example, the option `turkish` of the `babel` package uses the equal sign (`=`) as active shorthand character.

4 The `\xwcolorbox` and `\xwminipage` macros

TO MAKE it easier for users to create paragraph boxes and color boxes of texts and watermarks, the `xwatermark` package provides the macros `\xwcolorbox` and `\xwminipage`. The macro `\xwcolorbox` calls the macro `\xwminipage`. The `\xwminipage` macro is a `minipage` environment that may be used for framing watermarks. Like the `\watermarksetup` macro, these two macros are set with key-value lists as follows:

²The `fancyvrb` package has, e.g., `\DefineShortVerb[key=value pairs]{x}`.

```

100 \xwminipage{key=value list}
101 \xwcolorbox{key=value list}.

```

The `textcolor` in `\xwminipage` is the color of the text. In the case of `\xwcolorbox`, two color values are expected: `textcolor` and `fillcolor`. Texts with commas need to be enclosed in braces when submitted to these macros. The default values of the options of these macros are described in section [A.2](#).

These macros can call one/each other iteratively, e.g.,

```

102 \watermarksetup{fontfamily=txtt,fontseries=m,color=red,
103   align=center,scale=0.7,angle=0,xcoord=0,ycoord=0,
104   textmark={%
105     \xwminipage{width=\paperwidth,
106       text={%
107         \xwminipage{width=\paperwidth,align=left,
108           textcolor=magenta,text={\TeX\[\[.1ex] \LaTeX}
109         }%
110       \[\[1ex]
111       \xwminipage{width=\paperwidth,align=center,
112         textcolor=green,text={\TeX\[\[.1ex] \LaTeX}
113       }%
114     \[\[1ex]
115     \xwminipage{width=\paperwidth,align=right,
116       textcolor=orange,text={\TeX\[\[.1ex] \LaTeX}%
117   }}}}}

```

which is one of the examples in the file `xwatermark-examples.tex`.

The left and right watermarks on page 1 were produced with the following settings:

```

118 \watermarksetup{fontfamily=phv,fontsize=11pt,
119   fontseries=m,align=center,height=\paperheight,
120   width=\paperheight,angle=90,scale=1,xcoord=0,ycoord=-1,
121   textmark={%
122     \xwcolorbox{fillcolor=gray!55,width=\paperheight,
123       align=center,framesep=5pt,framerule=2pt,
124       framecolor=white,
125       text={\color{white}\framebox[2cm][c]{%
126         \phantom{Hello}}\hspace{1cm}\color{black}%
127         \textcolor{red}{\textbullet}\hspace{1cm}The marks %
128         on these pages were placed effortlessly %
129         by using the \textcolor{blue}{xwatermark} package

```

```

130     \hspace{1cm}\textcolor{red}{\textbullet}%
131     \hspace{1cm}\color{white}\framebox[2cm][c]%
132     {\phantom{Hello}}\rotatebox[origin=c]{90}{%
133     \makebox[1.4cm][c]{\phantom{Hello}}}%
134     }%
135   }%
136   \\\[17.2cm]
137   \xwcolorbox{framesep=5pt,framerule=2pt,
138   fillcolor=gray!55,framecolor=white,
139   align=center,width=\paperheight,
140   text={%
141     \color{white}\framebox[2cm][c]{\phantom{Hello}}%
142     \hspace{1cm}\color{black}\textcolor{red}%
143     {\textbullet}\hspace{4.3cm}User manual for the %
144     \textcolor{blue}{xwatermark} package\hspace{4.3cm}%
145     \textcolor{red}{\textbullet}\hspace{1cm}
146     \color{white}\framebox[2cm][c]{\phantom{Hello}}%
147     \rotatebox[origin=c]{90}{%
148     \makebox[1.4cm][c]{\phantom{Hello}}}%
149     }%
150   }%
151 }%
152 }

```

4.1 The `\ywcolorbox` macro

This macro has the same syntax and options as the `\xwcolorbox` macro except that the resulting `colorbox` is centered by using the `center` environment and the markup box `\makebox[0pt][c]{}`. It is intended for producing `colorboxes` such as the one for the abstract on 1. The abstract was produced with the following settings:

```

153 \ywcolorbox{framesep=5pt,framerule=2pt,framecolor=red!55,
154 align=justified,fillcolor=gray!25,width=\hsize,
155 text={%
156   \centering\xwcolorbox{align=center,fillcolor=white,
157   width=.5\hsize,text={\bfseries Abstract}}\\\[1ex]
158   The \styaxwatermark package enables the user ...
159 }%
160 }

```

Notice here that the macro `\ywcolorbox` calls the macro `\xwcolorbox` with its (`\ywcolorbox`'s) `framesep` and `framerule`.

Users may also like to consider the possible use of the `\fcolorbox` macro of the `xcolor` package in setting the textmarks.

5 Using `xwatermark` with the `geometry` package

SOME USERS have reported an apparent “conflict” between the `geometry` package and `xwatermark`, which according to them makes it rather unpleasant to properly position the watermarks. There is actually what, for want of a better term, I may refer to as a “mismatch” between `geometry` and `xwatermark` but certainly not a conflict. The problem emanates from the `geometry` package because it changes the scale, ratio, magnification, and other native dimensions of the paper to get the needed layout right all the time. The only layout parameter that the `geometry` package may retain is the paper center, which, unfortunately, does not always coincide with the text center. In fact, even the paperwidth and paperheight can be changed by the user of the `geometry` package.

Reconciling the `xwatermark` package with the `geometry` package at the very low level requires more effort than I can afford now. The interim solution is that the user should set his/her watermarks before loading the `geometry` package, or use the `geometry` package with the option `pass` in the preliminary runs (when setting the watermarks). The `pass` option is available from version 4.2 of the `geometry` package. It disables auto-layout and all of the `geometry` settings except `verbose` and `showframe`. It can be used to determine the page layout of the `\documentclass` and layouts created by other packages and manual settings. The user can also employ the option `showframe` of the `geometry` package to view how the scaling factors used by the `geometry` package might change native layout dimensions. The option `reset` of the `geometry` package is also useful in this regard.

The `geometry` package saves native (L)A_TE_X dimensions and switches in the macro `\Gm@dorg` before processing `geometry` package options. This macro is called by `geometry` when the options `pass` and `reset` are passed to it. Reconciling the two packages (`xwatermark` and `geometry`) at a high level will involve simply calling this macro within `xwatermark` whenever `xwatermark` detects that the `geometry` package has been loaded by the user. This is what has been done in the `xwatermark` package: the package has a boolean option called `dgeometry` (meaning “disable geometry”), which, if true, invokes the command `\Gm@dorg` of the `geometry` package to disable `geometry` package settings and enforce native paper layout dimensions. First the `xwatermark` package detects at the very last moment of the document preamble (just before `\begin{document}`) if the `geometry` package has been loaded by the user. If yes, and if the user has set `dgeometry=true` in the call to `xwatermark`, then `xwatermark` issues the command `\geometry{pass}`, which, as

mentioned earlier, calls `\Gm@dorg`.

After the effects of the `geometry` package are re-introduced (i.e., after setting `pass=false`), it might still be necessary to fine-tune the positions of the watermarks.

Because the `geometry` package stipulates that the command `\Gm@dorg` can be issued only in the document preamble, the switch `dgeometry` can appear only in `\usepackage{xwatermark}`, as in, e.g.,

```
161 \usepackage[printwatermark,allpages,dgeometry]{xwatermark},
```

but it matters not which of the two packages (`geometry` and `xwatermark`) is loaded first. To call `\Gm@dorg`, the `xwatermark` package uses the hook `\AtEndPreamble` from the `etoolbox` package. `\AtBeginDocument`, a native \LaTeX hook, is inapplicable in this case.

The user should also remember that he/she can set his/her own margins without using the `geometry` package, although many users find the `geometry` package more convenient.

6 Support for Unicode and UTF encodings

THE `xwatermark` package can be used with any font encoding, provided the `fontfamily` is properly declared before use. For example, with the following declarations on \XeTeX , Rembrandt Wolpert (wolpert@uark.edu) obtained some `.pdf` outputs that he is willing to share with other users:

```
162 \newfontfamily{\chinese}{STFangsong} % SinoType FangSong
163
164 \newcommand{\chttext}[1]{\chinese \XeTeXlinebreaklocale%
165 "jp" \XeTeXlinebreakskip=0pt plus 1pt #1}%
166 }
167
168 \watermarksetup{fontsize=5cm,align=center,
169 color=red!75!blue!25,angle=90,xcoord=-65,ycoord=-38,
170 scale=.49,textmark={=\fbox{\color{red!65}\chttext{%
171 watermark in Chinese or Japanese script}}=}%
172 }
173
174 \newfontfamily{\Gara}{Garamond Premier Pro}
175
176 \watermarksetup{fontsize=5cm,scale=.46,align=center,
177 angle=90,color=red!75!blue!25,xcoord=-72,ycoord=-38,
178 textmark={=\fbox{\color{red!65}\Gara The different %
```



```
179   ligature}=\[.35ex]}%
180 }.
```

It doesn't matter what the user declares as a `fontfamily` provided he/she declares it before using it and provided the declaration is valid. It is thus possible to mix scripts in one watermark (e.g., Latin, Chinese, Korean, Japanese, Arabic, Russian scripts, you name it).

7 Further examples of use of `xwatermark` package

THE FILE `xwatermark-examples.tex`, a source file of examples of use of the `xwatermark` package, together with its `pdf` version, are provided with this guide in the `xwatermark` bundle.

8 Future extension

CURRENTLY THE PACKAGE cannot put more than one watermark on the same page in a simple manner. To be able to do so, some tricks are necessary, as was done with the watermarks on the pages of this document. To position more than one mark on the same page, complicated tricks may be necessary, but are often possible. Markup boxes (especially zero-width boxes) can be handy in this respect, but the user should note the displeasure that unintended spaces can cause in boxes. It will be useful to relieve the user of such complications. This is an obvious track for extension of the package. One possible way to tackling this challenge is to accumulate the watermarks for each page before calling the shipout hooks, or loop over the watermarks (meant for the same page) within the shipout hooks, or write the watermarks to the auxiliary file and make more than one run to get them to the right pages.

The user should note the real meanings of saved-box commands (such as `\newsavebox`, `\savebox`, and `\usebox`) before using them. They are actually bins and not all declarations made outside the bins may be visible inside them.

There is also the issue with the `geometry` package mentioned in section 5.

9 Epilogue

BUG REPORTS and suggestions for further improvement and extension of the package are very much welcome.

10 Acknowledgement

I thank Joseph Wright (joseph.wright@morningstar2.co.uk) for his time in helping me resolve some of the challenges I encountered in coding this package—and for his continued support.

A Package options

WE CATEGORIZE the package options into global and local. Global options are those set either in `\documentclass` or in `\usepackage{xwatermark}` or with the macro `\watermarksetup`, while local options are those set with the macros `\xwminipage`, `\xwcolorbox` and `\ywcolorbox`.

A.1 Global options

The global package options are listed and described in table 1.

Table 1: Global options of the package

Option	Default	Meaning
<code>printwatermark</code>	<code>true</code>	Global boolean switch that determines whether watermark should be printed or not. This option can be passed to package as either <code>true</code> or <code>false</code> and can appear in the <code>\documentclass</code> options list or <code>\usepackage</code> . If for any reason you don't want the watermark printed in any run, you can enter <code>printwatermark=false</code> . If the option <code>draft</code> or <code>final</code> appears in the <code>\documentclass</code> or <code>\usepackage</code> , it won't affect the printing or otherwise of the watermark. The printing of the watermark is determined by the option <code>printwatermark</code> .

Continued on next page

Continued from last page

Option	Default	Meaning
allpages, evenpages, oddpages, pages, page		These determine the pages on which the watermarks are to be printed. The options <code>allpages</code> , <code>evenpages</code> , <code>oddpages</code> are boolean keys, while <code>pages</code> and <code>page</code> require values, e.g., <code>pages=x-x</code> and <code>page=x</code> . If the package is loaded and none of these options is passed to it, the default watermark (DRAFT) will be printed on the first page of the document and a warning message logged in the transcript file.
<code>textmark</code>	DRAFT	The text watermark.
<code>angle</code>	45°	The orientation of the watermark (text and/or picture).
<code>scale</code>	1	The scale of the text watermark.
<code>grayness</code>	0.8	The grayness of the text watermark. This is no longer available as an option since the grayness of the gray color can be set as a percentage (e.g., <code>color=gray!25</code>).
<code>color</code>	gray	The color of the text watermark.
<code>fontfamily</code>	cmss	The fontfamily of the text watermark.
<code>fontsize</code>	5cm	The fontsize of the text watermark. If for some reason you need other font sizes (e.g., 10pt, 11pt or 12pt for printing text watermarks in <code>\normalfont</code>), you will need to submit them as values of <code>fontsize</code> .
<code>fontseries</code>	b	The font series of the text watermark. If you need normal document text, put <code>fontseries=m</code> , which implies medium weight and width.

Continued on next page

Continued from last page

Option	Default	Meaning
xcoord	0	x-coordinate of watermark (with reference to the center of paper and not the text center).
ycoord	0	y-coordinate of watermark (coinciding with center of paper). Both <code>xcoord</code> and <code>ycoord</code> should be submitted without units, since the unit is set separately by <code>coordunit</code> .
coordunit	mm	The unit for x- and y-coordinates.
align	center	Alignment of the watermark (the key can assume the value <code>center</code> , <code>left</code> , <code>right</code> or <code>justified</code>). Any other alignment value is inadmissible and thus rejected by the package with a fatal error.
width	<code>\paperheight</code>	Width of <code>text</code> watermarks only; doesn't apply to graphic watermarks. For some design reasons we set <code>\paperheight</code> as the default value of <code>markwidth</code> instead of <code>markheight</code> .
height	<code>\paperwidth</code>	Height of text watermark.
picscale	1	Scale of picture watermark.
picbb	0 0 100 100	The bounding box (dvi mode) or viewport (pdf mode) of the picture watermark.
picfile		The filename of picture watermark. This, with its full path, must be submitted when including picture watermark.

Continued on next page

Continued from last page

Option	Default	Meaning
picfilex	eps/pdf	The filename extension of the picture watermark. Admissible extensions are <code>eps</code> , <code>pdf</code> , <code>png</code> and <code>jpeg</code> ; the latter three may be used in the case of pdf \TeX . The file extension should be passed without the dot. If the option is not passed to package, <code>xwatermark</code> selects <code>eps</code> (in dvi mode) or <code>pdf</code> (in pdf \TeX mode).
showcenter		Boolean that indicates if the center of the paper should be shown with a cross (and circle).
dgeometry	false	The package option that determines whether the page layout settings by the <code>geometry</code> package should be disabled so that the watermarks can be set more readily by <code>xwatermark</code> .

A.2 Local options

Local package options are those associated with the macros `\xwminipage`, `\xwcolorbox` and `\ywcolorbox`. They are described in table 2.

Table 2: Local options of the package

Option	Default	Meaning
<code>\xwminipage</code> macro		
width	<code>\paperwidth</code>	Width of the boxedminipage.
textcolor	black	Color of text inside boxedminipage.
framecolor	white	Color of frame of boxedminipage.
text	No text passed	Text that goes inside the boxedminipage.
align	center	Alignment of the text inside the boxedminipage of <code>\xwminipage</code> (expected values are <code>center</code> , <code>left</code> , <code>right</code> and <code>justified</code>).
<i>Continued on next page</i>		

Continued from last page

Option	Default	Meaning
framesep	3pt	\fboxsep of the boxedminipage.
framerule	0.4pt	\fboxrule of the boxedminipage.
\xwcolorbox macro		
width	\paperwidth	Width of the colorbox.
textcolor	black	Color of text inside colorbox.
framecolor	white	Color of frame of colorbox.
text	No text passed	Text that goes inside the colorbox.
align	center	Alignment of the text inside the colorbox of \xwcolorbox (expected values are <code>center</code> , <code>left</code> , <code>right</code> and <code>justified</code>).
framesep	3pt	\fboxsep of the colorbox.
framerule	0.4pt	\fboxrule of the colorbox.
\ywcolorbox macro		
The options of the macro \ywcolorbox are the same as those of the related macro \xwcolorbox. Refer to section 4.1 for the syntax of this macro.		