The `xwatermark` Package[☆]

Version 1.5.2

Ahmed Musa✉
Preston, Lancashire, UK

24th November 2011

Abstract

The `xwatermark` package puts user-supplied watermarks (graphics and/or arbitrary texts) on select pages of documents using user-friendly key-value interfaces. It has more functionality and dynamism than, for example, the packages `draftcopy`, `draftwatermark`, `watermark`, `draftmark`, `wallpaper`. More than one (graphics and/or text) watermark can be placed jointly or independently on the same document page or on select pages. Watermarks can be placed in the page background or foreground, and watermarks can conveniently be placed on select pages as rectangular or square tiles, depending on the user's choice. Some utility macros, namely, `\xwmminipage`, `\xwmcolorbox`, `\makecolobox` and `\fancypagenos` are also provided by the package for handy use in creating watermarks and for other uses. Watermarks (especially wallpapers) take their toll on computer resources, especially speed and save stack size. The packages in the `xwatermark` bundle (and beyond) have been optimized as much as currently possible. In many instances more than one run of the document will be needed to get the watermarks on the desired pages, especially if the user calls `\lastdocpage` to get the last page of the document.

[☆] The package is available at http://mirror.ctan.org/macros/latex/contrib/xwatermark/.

•   LICENSE   •

This work (i. e., all the files in the `xwatermark` manifest) may be distributed and/or modified under the conditions of the LaTeX 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. © MMXI

CONTENTS

1   INTRODUCTION

THE **XWATERMARK PACKAGE** puts user-specified watermarks (graphics and/or arbitrary texts) on select pages of documents. It has more functionality and dynamism than, for example, the packages `draftcopy`, `draftwatermark`, `watermark`, `draftmark` and `wallpaper` packages. The advantages of `xwatermark` package over these earlier packages include:

a) Both text and graphics watermarks are admissible within any watermark item or instance.
b) The user can dynamically customize the attributes (color, position, orientation, scale, the page(s)—first page, last page, all pages, odd pages, even pages, a particular page, and a

range of pages—on which the watermark should appear) of each watermark.

c) Watermarks can be placed in the background and in the foreground of document pages by simple instructions.

d) Rectangular and square wallpapers can be produced from watermarks to suit user needs without effort.

e) All the command options/keys are passed directly via user-friendly key-value interfaces, instead of being defined in the source file by several macros. There are only two main user commands: `\newwatermark` and `\newwallpaper`. The user is relieved of the need to remember and deploy several different macros, except, of course, that function keys are used. The list of keys and their default values for these functions are given in section 13.

With the `xcolor` package (not loaded automatically by the `xwatermark` 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 and printed as watermarks on the same page, and on different positions.

There are global and local package options. These are listed and explained in section 13.

Users who have since complained of not being able to conveniently place more than one watermark on the same page can now heave a sigh relieve: the current version of the package has enabled this functionality. You can now mix text and graphics watermarks and wallpapers and place as many of them as you like on the same page. This version of the package comes with optimized looping macros and a key management system (the `ltxkeys` package) to enable several watermarks and wallpapers to be placed efficiently on the same document pages. The `ltxkeys` package can be used for general key parsing.

2   USER INTERFACES

2.1   Loading the package

In style files the package may be loaded with `\RequirePackage` and in document files with `\usepackage` together with the package keys that can be passed as options.

Note 2.1 Some of the keys are 'option keys,' i.e., they can appear only in `\documentclass` or `\usepackage` and not as, or in, arguments of other functions or macros. The 'non-option keys' are those that can't appear in `\documentclass` or `\usepackage` but in the arguments of other macros. If a key is a non-option key and the user submits it to `\documentclass` or `\usepackage`, the package will alert the user. The same thing can be expected when a key is an option key and is submitted outside of `\documentclass` or `\usepackage`. The 'need value' keys are keys that can't be called without a user-supplied value.

The package keys `printwatermark` and `disablegeometry` are option keys, and hence can be called as follows:

```
                            ┌──── Package loading ────┐
1    \RequirePackage[printwatermark,disablegeometry]{xwatermark}
2    \usepackage[printwatermark,disablegeometry]{xwatermark}
```

The other options may be submitted via user commands like `\newwatermark` and `\newwallpaper`. Please see Tables 1 and 2 for a full listing of all the available package and command options. By design, the boolean option `printwatermark` should not appear in the macros `\newwatermark` and `\newwallpaper` but as a package or `\documentclass` option. It is disabled just before `\begin{document}` and any attempt to pass it via `\newwatermark` or `\newwallpaper` 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.

Note 2.2 When your watermark is not printed, first check that the option `printwatermark` is `true`. This is one of the means to control the printing of watermarks. The others are through the following commands (more details are available in subsection 3.5):

```
                      \dummywatermark, \DiscardAllWatermarks, etc
3    \dummywatermark, \DiscardAllWatermarks, \UseDummyWatermarks,
4    \DiscardDummyWatermarks
```

The option `textmark` implies text watermarks, for which all the font properties can be selected. It does not apply to graphics watermarks. For graphics watermarks you need the keys: `picfile` (the graphics/picture filename, with its full path but without its extension), and `picfileext` (the file extension). Admissible file extensions are `ps`, `eps`, `pdf`, `png`, `mps` and `jpeg`; they should be submitted without the dot. The extensions `ps` and `eps` are for `dvi` files, while the rest are for `pdf` runs. Additional information is needed (see section 5)[★1].

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

a) The value of the `textmark` may be any arbitrary multi-line text, such as

```
                          Example: textmark
5    textmark=Hello world,\\[.25\baselineskip] We're here.
```

b) 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 as values of graphics keys). The package has a user-friendly interface for inserting graphics watermarks and wallpapers, which does not require the user to directly employ `\includegraphics`.

c) The `textwidth` and `picwidth` 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`, whose default value is `\paperwidth`.

d) 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 the `textscale` (or `picscale`) options will have to be suitably chosen.

The boolean options `firstpage`, `lastpage`, `allpages`, `firstpage`, `oddpages` and `evenpages`, which specify the pages that should receive watermarks, may be replaced by any of the options `page=x`, `pages=x-y`, `pagex={x,y,z}`, where 'x,' etc., stand for any page number. If you enter, for example, `pages=0-10`, all pages from 1 to 10 will receive the 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 and `printwatermark` is true, 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-y` as option to package, don't forget to include the equality sign (`=`), otherwise the option will trigger an 'unknown option/key' error. The key `pages` expects a page range with the pages separated by a hyphen, while `pagex` expects a comma-separated list of pages. For obvious reasons, the value of the key `pagex` must always be given in balanced curly braces.

When specifying package options either in `\usepackage` or `\documentclass` (or indeed in the macros `\newwatermark`, `\newwallpaper`, `\xwmminipage`, `\xwmcolorbox` and `\fancypagenos`), the

---

[★1] When the options `align`, `height`, `width`, `angle`, `scale`, `xpos`, `ypos` and `color` appear without prefixes such as `pic` or `text`, they refer to the text watermark and not the graphics watermark. The user can thus use these options in place of `textalign`, `textheight`, `textwidth`, `textangle`, `textscale`, `textxpos`, `textypos` and `textcolor`,

following points should be noted:

a) Multiple lines are permitted but not blank lines.
b) Extra paces between options and words are ignored.
c) Active characters (those of catcode 13) may be allowed (but see subsection 6.10 for further comments).
d) Options are mostly order-agnostic, except `graphicsoptions`, whose values take precedence over those supplied via other keys (see subsection 5.1).

The global boolean option `printwatermark=true` (or `=false`) should ideally be set when loading the package, e. g.,

---
Package option: printwatermark

```
6    \usepackage[printwatermark]{xwatermark}
```

or in the `\documentclass` options list:

Package option: printwatermark

```
7    \documentclass[a4paper,12pt,printwatermark]{article}
8    \usepackage{xwatermark}.
```

The remaining options should ideally be set dynamically using the macro `\newwatermark` or `\newwallpaper`. These other options can be set for each page, as on the pages of the accompanying example files.

3   THE `\NEWWATERMARK` MACRO

The use syntax for the command `\newwatermark` is as follows

New macro: \newwatermark

```
9    \newwatermark[⟨keyval⟩]{⟨mark⟩}
10   \newwatermark*'[⟨keyval⟩]{⟨mark⟩}
```

where ⟨`keyval`⟩ is the list of keys and their values (called the watermark attributes) and ⟨`mark`⟩ is the text watermark. Graphics watermarks are to be specified with their file name, file extension, etc. The full lists of the available keys for the macro `\newwatermark` and others are available in section 13.

The starred (⋆) variant of `\newwatermark` puts the watermark in the foreground instead of the background, and the prime (') variant is ignored, i. e., no watermark is produced (see subsection 3.5).

The macro `\newwatermark` can be used as in

Example: \newwatermark

```
11   \newwatermark[pagex={2,5,7},fontfamily=bch,color=gray!25,angle=45,scale=3,
12     xpos=0,ypos=0]{DRAFT},
```

where the `textmark` has been enclosed in curly braces as the last argument of the macro. The

---

respectively. However, options referring to graphics watermarks must always be prefixed with `pic` (e. g., picfile).

options (called the watermark attributes) are expected in square brackets. The `textmark` (which is 'DRAFT' in the above example) can also be given within square brackets, in which case the curly braces will be empty:

> **Example: \newwatermark**
>
> ```
> 13    \newwatermark*[page=10,fontfamily=bch,color=gray!25,angle=45,scale=3,xpos=0,
> 14       ypos=0,textmark=DRAFT]{}.
> ```

The option `printwatermark` may appear in only `\usepackage` or `\documentclass` options list, since it is disabled at `\begin{document}`. However, the options `firstpage`, `lastpage`, `allpages`, `oddpages` and `evenpages`, etc., which specify watermark pages, can and should appear in the command `\newwatermark`. 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 it 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 `\newwatermark` macro with the page option `page=⟨no⟩` in the preamble of your document after issuing

> **Examples: \usepackage, printwatermark**
>
> ```
> 15    \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. In fact, you can collect all the watermarks in the document preamble or in a configuration file with the command `\newwatermark`.

Note 3.1 Each call to `\newwatermark` must contain the page(s) that will receive the watermark(s), otherwise the user will be alerted. The page specifiers are:

> **page specifiers**
>
> ```
> 16    page=x, pages=x-y, pagex={x,y,z}, firstpage, lastpage, allpages,
> 17    oddpages, evenpages
> ```

3.1 Options without values

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

> **Example: \newwatermark**
>
> ```
> 18    \newwatermark[firstpage,fontfamily=,color=gray!25,angle=45,scale=0.8,
> 19       xpos=0,ypos=0,textmark=]{},
> ```

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 empty textmark is valid and implies that no watermark should be printed. The absence of `fontfamily` in 'fontfamily=' will compel (LA)TEX 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).

3.2 Emptying the watermarks of some pages or objects

If you issue any of the statements

```
                          ┌─── Example: page specifiers ───┐
20    page=x, pages=x-y, pagex={x,y,z}, firstpage, lastpage, allpages=true,
21    evenpages=true, oddpages=true
```

together with printwatermark=true but you don't want the mark on any particular page, we can simply set \newwatermark[other keys,textmark=]{} or, to the same effect, we may set \newwatermark[other keys]{}, where 'other keys' may include the page specifiers[★2]. These both imply that the text watermark for the given page is empty. This can be useful when transiting from one watermark type to another. Moreover, since both picture and text marks can be submitted via one and the same command \newwatermark (see subsection 3.3), this technique may be used to empty the text watermark for the given page or range of pages. For example,

```
                           ┌─── Example: \newwatermark ───┐
22    \newwatermark[allpages,fontfamily=put,color=white,fontsize=3cm,scale=1,
23       picbb=112 619 242 751,picscale=3,picfile=./graphics/myfig,picfileext=eps,
24       width=\paperheight,align=center,angle=0,xpos=0,ypos=0]{}
```

will print only the picture watermark, since the textmark is empty here.

3.3 Printing both picture and text watermarks on same page

Both picture and text marks can be submitted and printed on the same page via one and the same \newwatermark. For example,

```
                           ┌─── Example: \newwatermark ───┐
25    \newwatermark[pages=1-2,fontfamily=put,color=white,fontsize=3cm,scale=1,
26       picbb=112 619 242 751,picscale=3,picfile={./graphics/myfig},
27       picfileext=eps,width=\paperheight,align=center,angle=0,xpos=0,
28       ypos=0]{Hello World}
```

However, both the picture and text marks will then share the same subset of the attributes (position, angle, align, etc.). When text and graphics watermarks appear on the same page, the recommended approach is to submit the two types of watermark by two separate calls to \newwatermark.

3.4 The usefulness of the white color

You can deploy the white color to great effect in designing text watermarks. Also, 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 \newwatermark 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 \newwatermark[other keys,textmark=]{} or remove (or comment out) the \newwatermark command for that (or indeed any) page. See also subsection 3.5.

3.5 Dummy watermarks

When you don't need a watermark printed, you can simply replace its \newwatermark with \dummywatermark, instead of commenting out the entire watermark or using color=white. Both

---

[★2] In the case of graphics watermarks, setting \newwatermark[other keys,picfile=]{} will prompt a 'no file' error.

the macros \newwatermark and \dummywatermark have the same syntax and expect the same
number and types of arguments:

```
                              New macro: \dummywatermark
29    \dummywatermark[pages=12-13,fontfamily=phv,fontsize=11pt,fontseries=m,
30      align=center,height=\paperheight,width=\paperwidth,angle=90,scale=1,
31      xpos=0,ypos=-1
32    ]{Example}
```

And when you don't want any of your watermarks printed, you could simply issue the option
printwatermark=false or call the command \DiscardAllWatermarks. These will simply turn
all instances of \newwatermark command into \dummywatermark. In any run, you may de-
cide to use some or all of the dummy watermarks. To use all dummy watermarks, you issue
the command \UseDummyWatermarks before the instances of \dummywatermark. To again dis-
regard all subsequent dummy watermarks, which is the default state, simply call the command
\DiscardDummyWatermarks. These commands provide a convenient scheme for deciding the wa-
termarks to be printed with minimal typing. For wallpapers, there is the corresponding command
\dummywallpaper. Also, putting a prime sign (') on \newwatermark or \newwallpaper turns the
command into a dummy mark, but only for that single instance. Subsequent \newwatermark and
\newwallpaper without primes will produce watermarks and wallpapers, respectively.

4   WALLPAPERS

4.1   The \newwallpaper macro

The command \newwallpaper can be used to produce rectangular and square tiles on document
pages. The use syntax for the command \newwallpaper is

```
                              New macro: \newwallpaper
33    \newwallpaper[⟨keyval⟩]{⟨mark⟩}
34    \newwallpaper*'[⟨keyval⟩]{⟨mark⟩}
```

where ⟨keyval⟩ is the list of keys and their values (called the attributes) and ⟨mark⟩ is the text
(and not graphics) watermark. Graphics watermarks are again to be specified with their file name,
file extension, etc. The full lists of the available keys for the macro \newwallpaper are available
in Table 2.

The starred (⋆) variant of \newwallpaper puts the watermark in the foreground instead of the
background, and the prime (') variant is ignored, i. e., no wallpaper is produced (see subsection 3.5).

When you get unexpected tiles, you first should consider enabling or disabling the keys squaretiles
(default true) and/or boxalign (default center). The key boxalign may assume one of the values
in the set t-l, t-r, b-l, b-r, s or top-left, top-right, bottom-left, bottom-right, center,
justified.

5   GRAPHICS WATERMARKS

For graphics/picture watermarks, you need the picfile (the graphics filename, with its full path
but without its extension), picfileext (the picture filename extension without the dot), picbb

(the picture bounding box), and `picscale` (the picture scale)[★3]. 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 is running or not (normal extensions are `eps` for dvi mode and `pdf` for PDFTEX mode). In fact, the package does search hard on the given paths for other admissible file types with the base filename the user has specified. If you have the graphics file in both `eps` and PDF-compatible 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 (PDF or `dvi`) in which it is running.

5.1   Passing key values to `\includegraphics` directly

The `xwatermark` package uses `graphicx` package's `\includegraphics` to insert graphics watermarks. Users can pass valid key values to the command `\includegraphics` directly via the macros `\newwatermark` and `\newwallpaper`. The key to use for this purpose is `graphicsoptions`. The following points should be noted in respect of the key `graphicsoptions`:

  a) Values of the key `graphicsoptions` must always be enclosed in curly braces, since they are expected to be more than one.

  b) Only keys and values valid for the command `\includegraphics` may appear in the command `graphicsoptions`. Valid keys for `\includegraphics` are

```
┌──────────────── \includegraphics keys ────────────────────┐
35   bb, bbllx, bblly, hiresbb, viewport, trim, height, width, natheight,
36   natwidth, totalheight, angle, origin, keepaspectratio, scale, clip,
37   draft, type, ext, read, command
└───────────────────────────────────────────────────────────┘
```

  c) Key values submitted via `graphicsoptions` supersede those submitted outside it, even if those outside `graphicsoptions` appear earlier than `graphicsoptions` in the command `\newwatermark` or `\newwallpaper`.

  d) Values submitted via `graphicsoptions` have only local effect, in the sense that they become null and void outside of `\newwatermark` or `\newwallpaper`. If the user wants the key values submitted via `graphicsoptions` to prevail for all subsequent watermarks and wallpapers, then he should use the command `\GraphicsOptions`. Values passed via `\GraphicsOptions` don't only have global effect, but they always override those submitted via `graphicsoptions`.

It should be noted that `\GraphicsOptions` isn't a key but a stand-alone command with the following syntax:

```
┌──────────────── New macro: \GraphicsOptions ──────────────┐
38   \GraphicsOptions{⟨keyval⟩}
└───────────────────────────────────────────────────────────┘
```

where ⟨`keyval`⟩ are admissible keys for the command `\includegraphics` and their user-supplied values. The values so suggested by `\GraphicsOptions` override those given via the keys of the `xwatermark`, including `graphicsoptions`. Such values remain in force until they are changed later by another call to `\GraphicsOptions`.

An example follows:

```
┌──────────────── Example: graphicsoptions ─────────────────┐
39   \newwallpaper[%
40     page=10,picangle=45,tilexoffset=0pt,tileyoffset=0pt,picontoptext=false,
└───────────────────────────────────────────────────────────┘
```

[★3] These options have longer, easier to remember, names; see Table 2.

```
41      boxalign=top-left,picbb=116 428 477 718,picscale=2,picfile=tabu-test1,
42      tileno=4,picfileext=pdf,graphicsoptions={clip,keepaspectratio,hiresbb}
43    ]{mypicture}

44    % or globally as
45    \GraphicsOptions{clip=true,keepaspectratio,hiresbb}
```

The commands \DeclareGraphicsExtensions and \DeclareGraphicsRule of the graphics package can still be invoked before setting graphics watermarks.

5.2   The graphics input paths

Users can suggest the possible locations of the graphics watermarks to the package by using the command \watermarkpaths, whose syntaxes are

```
                        New macro: \watermarkpaths
46    \watermarkpaths[⟨pre⟩](⟨post⟩){{path-1}{path-2}...{path-n}}
47    \watermarkpaths*[⟨pre⟩](⟨post⟩){path-1,path-2,...,path-n}
```

Here, ⟨pre⟩ and ⟨post⟩ are optional arguments that apply to all the given paths. Caution should be exercised when using these optional arguments, since when used incorrectly they can yield the wrong path (see the example below). In the unstarred variant all the paths must be provided in surrounding curly braces and must have no commas, otherwise the package will raise an error. The starred (⋆) variant expects paths to be separated by commas. The package works hard to find your watermark on the suggested path.

```
                        Examples: \watermarkpaths
48    \watermarkpaths{{./}{./graphics/}{./graphics/recentfiles/}}
49    \watermarkpaths*{./,./graphics/,./graphics/recentfiles/}
50    \watermarkpaths(/){{.}{./graphics}{./graphics/recentfiles}}
51    \watermarkpaths*(/){.,./graphics,./graphics/recentfiles}
52    % Note the empty balanced braces below. Without them, the first entry
53    % (which is supposed to be {./}) will be wrong:
54    \watermarkpaths[.](/){{}{/graphics}{/graphics/recentfiles}}
55    \watermarkpaths*[.](/){{},/graphics,/graphics/recentfiles}
```

By default, the packages works hard to preserve outer curly braces, unless and until they are required to be removed.

The command \watermarkpaths inherits the current contents of LaTeX's \input@path command and graphics package's \Ginput@path (the latter takes argument from \graphicspath).

6   OTHER ASPECTS OF PACKAGE ARCHITECTURE AND USE

6.1   \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 option 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

---

┌──────────────── Example: package loading ────────────────┐
56 │ \usepackage[printwatermark=false]{xwatermark}, │
└──┘

and vice versa. It should, however, be noted that some package options and keys are restricted either to the \documentclass and \usepackage (this applies to the so-called 'option keys') or to the various user macros (in the case of 'non-option keys'). Normally, the package will alert the user to the wrong call of any of the options and keys.

If you don't need the watermark on any page of your document, simply replace the option printwatermark (=true) with printwatermark=false in \usepackage or \documentclass. 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 use the tools of subsection 3.5.

6.2 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.

6.3 The coordinates of the watermark

The watermark coordinates (specified by xpos and ypos) 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 changing the value of coordunit. For example,

┌──────────── Examples: \newwatermark, coordunit ────────────┐
57 │ \newwatermark[other attributes,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), and em (width of capital M).

6.4 Wrong location of the watermark

If you discover that the watermark is wrongly positioned on the page(s) of your document, as some users have had course to complain, the chances are that you have submitted wrong coordinates (values of xpos and ypos) to the package or the watermark's width (textwidth or picwidth) 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 the anomaly, no attempt has been made in the package to warn users in this regard. If you do not specify the keys xpos and/or ypos at all in the call to \newwatermark, their default values will be used by the package. Also, if you list these keys without their values in the call to \newwatermark, their default values (xpos=0 and ypos=0, which yield the center of paper) will be assumed by the package. The default value of the watermark's width is \paperheight, and not \paperwidth as might be expected.

When the geometry package is loaded together with the xwatermark package, page layout scale changes by the geometry package may result in the watermarks being positioned slightly away from the intended position. See section 9 for further details.

6.5 Wrong size of the watermark

When you discover that your text or graphics watermark is not of the size you expect, then you

should check the global and local scale and width of the watermark. It is most likely that the chosen combination is wrong or inconsistent. Global and local package options are described in section 13. For example, choosing `scale=0.7` and `width=\paperwidth` may yield something unexpected. So will mixing inconsistent global and local scales or width, or both.

6.6 Breaking the watermark into lines

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

```
Example
58  \newwatermark[evenpages,fontfamily=ptm,angle=45,scale=.7,
59    align=center,color=green,xpos=0,ypos=0]{Directorate\\[.25ex]Only}

60  \newwatermark[allpages,fontfamily=ptm,angle=45,scale=.8,align=left,
61    color=green,xpos=0,ypos=0]{Control\\[.25ex]Version}.
```

More complex examples are available in the example source and PDF files that shipped with this package.

6.7 The alignment of the watermark

The alignment of the watermark is controlled by the keys `align`, `textalign` and `boxalign`. The first two are equivalent and 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. The admissible values for the key `boxalign` are given in Table 2.

6.8 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 `showpagecenter`, which may be issued (dynamically for each page) with the `\newwatermark` macro as follows:

```
Example: showpagecenter
62  \newwatermark[allpages,showpagecenter]{}
63  \newwatermark[page=1,showpagecenter=true]{}

64  \newwatermark[allpages,showpagecenter,fontfamily=ptm,angle=60,scale=.7,
65    color=brown!25!yellow!75,coordunit=cc,xpos=0,ypos=0]{Confidential!}.
```

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

```
Example
66  \newwatermark[pages=1-2,showpagecenter=false]{}

67  \newwatermark[page=10-\lastdocpage,showpagecenter=false,fontfamily=panr,
68    angle=60,scale=.7,color=brown!25!yellow!75,coordunit=cc,xpos=0,ypos=0]
69    {Confidential!}
```

6.9   The last page of the document

You can easily obtain the last page of the document with the label xwmlastpage, which is automatically provided by the package: the user doesn't have to insert it himself. In general, you can use the command \xwmgetpagenumber to extract page numbers from LaTeX labels (even in expansion contexts). More than one run may be necessary in extracting page numbers from this command. The following example inserts the watermark from second to the last page to the last page. Note that in this example the starting page is necessarily enclosed in curly braces so as to distinguish the two hyphens that serve different purposes.

```
Example: \xwmgetpagenumber ,\lastdocpage
70   \newwatermark[pages={\lastdocpage-2}-\lastdocpage,angle=90,
71     scale=1,xpos=0,ypos=-1]{This is page \thepage~of~\pageref*{xwmlastpage}}
```

The command \lastdocpage is equivalent to \xwmgetpagenumber{xwmlastpage}.

6.10   Active characters

Active characters (i.e., those of category 13) and expandable commands can normally be used as values of the textmark key in the \newwatermark macro. However, such values cannot be passed via the \documentclass or the \usepackage{xwatermark} command without first loading one of the packages: xkvltxp, kvoptions-patch and catoptions packages. That is, the following should work:

```
Example
72   \RequirePackage{catoptions}
73   \documentclass[myoption={My watermark,\\[2ex]
74     designed~by \textsc{Mr.~J\"ohnson}}]{class-file}

75   \begin{document}
76   Blackberry lily ...
77   \end{document}
```

In plain TeX the only active character is the tie character (i.e., \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 active[★4]. The user can use such active characters in values of the textmark key without locally changing their catcode to 11 or 12. 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.

7   MACROS FOR CREATING BOXES

To make it easier for users to create paragraph boxes and color boxes of texts and watermarks, the xwatermark package provides the macros \xwmcolorbox and \xwmminipage. The macro \xwmcolorbox calls the macro \xwmminipage.

7.1   The \xwmminipage macro

The \xwmminipage macro is a minipage environment that may be used for framing watermarks.

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

It accepts verbatim material. Like the \newwatermark macro, this macro is called with key-value pairs as follows (see Table 2 for a full listing of the available keys):

```
New macro: \xwmminipage

78   \xwmminipage[key=value list]{balanced text}
79   \xwmcolorbox[key=value list]{balanced text}.
```

The textcolor key in \xwmminipage is the color of the text. In the case of \xwmcolorbox, four color values are expected: textcolor, fillcolor, outerframecolor and innerframecolor. Texts with commas need to be enclosed in braces when submitted to these macros. The default values of the keys of these macros are described in subsection 13.2.

The macros \xwmminipage and \xwmcolorbox can be nested within and among themselves, e. g.,

```
Example: \xwmminipage

80   \newwatermark[pagex={1,3,10},fontfamily=txtt,fontseries=m,color=red,
81     align=center,scale=0.7,angle=0,xpos=0,ypos=0]{%
82       \xwmminipage[width=\paperwidth]{%
83         \xwmminipage[width=\paperwidth,align=left,textcolor=magenta]
84           {\TeX\\[.1ex] \LaTeX}\\[1ex]
85         \xwmminipage[width=\paperwidth,align=center,textcolor=green]
86           {\TeX\\[.1ex] \LaTeX}\\[1ex]
87         \xwmminipage[width=\paperwidth,align=right,textcolor=orange]
88           {\TeX\\[.1ex] \LaTeX}%
89       }%
90   }
```

More complicated examples can be found in the example files. But some of the complications found in the example files are unnecessary since several simple watermarks can be placed on the same document page by specifying the same page number for those simple watermarks.

7.2 The \xwmcolorbox macro

The macro \xwmcolorbox provides a key-value interface to xcolor package's \fcolorbox[5].

```
New macro: \xwmcolorbox

91   \xwmcolorbox[key=value list]{balanced text}.
```

7.3 The \makecolorbox macro

This macro has the same syntax and options as the \xwmcolorbox 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 article abstracts. The user may experiment with the following settings, from which the abstract of this guide was produced:

---

[5] The xwatermark package also comes with the macro \xwmshade which is similar to \xwmcolorbox, but which, unlike \xwmcolorbox, can break neatly across pages (in the manner of the framed package). But since no watermark is expected to break across pages, the macro \xwmshade isn't described in this guide. Power users should still be able to use it.

```
        New macro: \makecolorbox
92   \makecolorbox[framesep=4pt,framerule=1pt,innerframecolor=red!55,
93     outerframecolor=ForestGreen,align=justified,
94     fillcolor=gray!25,width=.95\hsize,boxalign=center,
95     height=2.5mm,depth=0mm,framebox]{%
96       \centering\xwmcolorbox[align=center,fillcolor=white,
97       innerframecolor=blue,outerframecolor=orange,width=.5\hsize,
98       height=2mm]{\textbf{Abstract}}\\[\baselineskip]
99       The \pkg'{xwatermark} provides facilities for \ldots
100  }
```

Notice here that the macro \makecolorbox calls the macro \xwmcolorbox.

8   THE \fancypagenos MACRO

The macro \fancypagenos, which has the following syntax, can be used to position and format page numbers in the desired fashion. Its keys and their default values are described in Table 2. Page numbers produced by \fancypagenos will, by default, appear in the foreground, so that they can be seen on top of watermarks. If you want the page numbers to appear in the background, then set sendtoback=true as one of the key-value pairs in the call to \fancypagenos.

```
          New macro: \fancypagenos
101   \fancypagenos[key=value pairs]
```

Even after issuing the command \fancypagenos, you can still decide not to print the fancy page numbers by calling the command \NoFancyPageNumbers. The complement of the command \NoFancyPageNumbers is \FancyPageNumbers.

9   USING XWATERMARK WITH GEOMETRY PACKAGE

Because the geometry package changes the scale, ratio, magnification, and other native dimensions of the paper to get the needed layout right all the time, the geometry package may interfere with the xwatermark package. 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.

Feasible solutions to this problem include setting the watermarks before loading the geometry package; using the geometry package with the option pass in the preliminary runs, when setting the watermarks (see further details below); using true dimensions (e. g., coordunit=truept); and using relative, rather than absolute, dimension units (i. e., em and ex). The power-user can also experiment with the primitives \magnification, \mag and \magstep.

The pass option of the geometry package has been available from version 4.2 of the geometry package onwards. It disables auto-layout and all of the geometry package 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 (LA)TEX 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, geometry) at a 'high

level' will involve simply calling this macro within the xwatermark package 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 disablegeometry, 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 suggested disablegeometry=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 the xwatermark package option disablegeometry=false), it might still be necessary, depending on the user need, 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 disablegeometry can appear as option only in \documentclass or \usepackage{xwatermark}. But it matters not which of the two packages (geometry, xwatermark) is loaded first. To call \Gm@dorg, the xwatermark package uses the hook \BeforeStartOfDocument from the catoptions package. \AtBeginDocument, a native LATEX hook, is inapplicable in this case.

10   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 XƎTEX, Rembrandt Wolpert (wolpert@uark.edu) obtained some .pdf outputs that he is willing to share with other users:

```
                        ┌─ Example: \newfontfamily ─┐
102   \newfontfamily{\chinese}{STFangsong} % SinoType FangSong

103   \newcommand{\chtext}[1]{{\chinese \XeTeXlinebreaklocale "jp"
104     \XeTeXlinebreakskip=0pt plus 1pt #1}%
105   }

106   \newwatermark[allpages,fontsize=5cm,align=center,
107     color=red!75!blue!25,angle=90,xpos=-65,ypos=-38,scale=.49]
108     {=\fbox{\color{red!65}\chtext{watermark in Chinese or Japanese script}}=}

109   \newfontfamily{\Gara}{Garamond Premier Pro}

110   \newwatermark[allpages,fontsize=5cm,scale=.46,align=center,
111     angle=90,color=red!75!blue!25,xpos=-72,ypos=-38]
112     {=\fbox{\color{red!65}\Gara The different ligature}=\\[.35ex]}.
```

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).

11   REPEATED GRAPHICS IN A DOCUMENT

For graphics watermarks, the watermark image, or any other image that is repeated in the document, has the potential to make the processed version of the document surprisingly large. The

problem is that the default mechanisms of graphics usage add the image at every point it is to be used, and when processed, the image appears in the output file at each such point.

See the UKTUG FAG, version 3.20 (2010), entry number 146, page 95, for the available solutions to this problem. As described by this reference, if the PostScript version of the file is destined for conversion to PDF, either by a ghostscript-based mechanism such as ps2pdf or by, for example, Acrobat Distiller, the issue is not as important, since the distillation mechanism will amalgamate graphics objects whether or not the PostScript has them amalgamated. PDFTEX does the same job with graphics, automatically converting multiple insertions into pointers to graphics objects. See also the \pdfxform command and instructions about XObjects in PDFTEX user manual.

12 FURTHER EXAMPLES OF USE OF XWATERMARK PACKAGE

The files xwatermark-examples1.tex and xwatermark-examples2.tex, source files of examples of use of the xwatermark package together with their PDF versions, are provided with this guide in the xwatermark bundle.

13 PACKAGE OPTIONS AND MACRO KEYS

We categorize the package options and keys into global and local. Global options are those set either in \documentclass or in \usepackage, while local options are those set with the macros \newwatermark, \newwallpaper, \xwmminipage, \xwmcolorbox and \makecolorbox.

13.1 Global options

The global package options are listed and described in the following Table 1.

Table 1: Global options of the package

All the keys in this table are 'option keys,' i.e., they can appear only in \documentclass or \usepackage. If a key is an option key and it is issued in functions other than \documentclass or \usepackage, the package will alert the user to that effect.

Option	Default	Meaning
defaultfirstpage, default-first-page	1	The default first page of document, when the first page hasn't been known yet.
defaultlastpage, default-last-page	1	The default last page of document, when the last page has't been shipped out yet.
draft	false	The document is in draft mode.[See note 1.1]
final	true	The document is in final mode.
disablegeometry, disable-geometry	false	This option determines whether the page layout settings by the geometry package should be disabled so that the watermarks can be set more readily by xwatermark.[1.2]
frontpagestyle, front-page-style	empty	The style of the front page, in the sense of the fancyhdr package.
picontoptext, pic-on-top-text, picture-on-top-text	true	The boolean that determines if picture watermark is placed on top of text watermark or otherwise, when they occur at the same location on a page.
printwatermark, print-watermark	true	The global boolean switch that determines whether watermarks should be printed or not.[1.3]

Continued on next page

Continued from last page

Option	Default	Meaning
`resetpaperorigin`, `reset-paper-origin`	`false`	The boolean to call to reset the paper origin (`\pdfhorigin`, `\pdfvorigin`, `\hoffset` and `\voffset`) to zero unit.
`showpagecenter`, `show-page-center`	`false`	Boolean that indicates if the center of the paper should be shown with a cross (and circle).[1.4]
`textontoppic`, `text-on-top-pic`, `text-on-top-picture`	`false`	The reverse of `picontoptext`.
`usedummymarks`, `use-dummy-marks`	`true`	The boolean that instructs `xwatermark` to print watermarks that have been defined by the `\dummywatermark` or `\dummywallpaper` in addition to those defined by `\newwatermark` and/or `\newwallpaper`.
`watermarkparser`, `watermark-parser`	`;` (semicolon)	The option determines the watermark parser, a quantity that is used internally to build watermarks into lists.[1.5]

Table 1 notes

[1.1] `draft` and `final` are complementary (biboolean) options, i.e., when one is true, the other is automatically set to false.

[1.2] Version 5.6 and higher of the `geometry` package make this option less likely to be needed, unlike the earlier versions of `geometry`.

[1.3] This option can be passed to package as either `true` or `false` and can appear in the `\documentclass` options list or `\usepackage`. If for any reason you don't want the watermark printed in any run, you can enter `printwatermark=false`. If the option `draft` or `final` appears in the `\documentclass` or `\usepackage`, it won't affect the printing or otherwise of the watermark, apart from determining if graphics watermarks are actually inserted instead of framed empty boxes. The printing of the watermark is determined by the option `printwatermark`.

[1.4] The `xwmgrid` package, which provides a full gridding functionality consistent with the `xwatermark` package, will be released shortly.

[1.5] If you do have the `watermarkparser` in your watermark, it must be enclosed in curly braces, otherwise the package will flag a fatal error. You can change the `watermarkparser` to say '`|`' (vertical bar), in case you have too many '`;`' (semicolons) in your watermarks and you are fed up with enclosing them in braces.

13.2 Local options

Local package options are those associated with the commands `\newwatermark`, `\newwallpaper`, `\xwmminipage`, `\xwmcolorbox`, `\makecolorbox` and `\fancypagenos`. They are described in the following Table 2.

Table 2: Local (function-dependent) options of the package

Option	Default	Meaning
Most of the keys in this table are 'non-option keys,' i.e., they can't appear in `\documentclass` or `\usepackage` but in the arguments of the given functions. If a key is a non-option key and the user submits it to `\documentclass` or `\usepackage`, the package will alert the user.		
\newwatermark macro		
`angle`, `textangle`, `text-angle`	$0°$	The orientation of text watermark.

Continued on next page

Continued from last page

Option	Default	Meaning
align, textalign, text-align	center	The internal horizontal alignment of the watermark within a watermark box.[See note 2.1]
page, pages, pagex, firstpage, lastpage, allpages, evenpages, oddpages		These determine the pages on which the watermarks are to be printed.[2.2]
boxalign, box-align	center/s	The alignment of the watermark box.[2.3]
coordunit, coord-unit, position-unit	mm	The unit for x- and y-coordinates of watermark.
color, textcolor, text-color	gray!25	The color of the text watermark.
draft	false	This option will insert framed empty boxes in place of graphics watermarks.
draftboxcolor, draft-box-color	blue	The color of the box that is printed in place of graphics watermarks in draft document mode.
final	false	This option actually does nothing for now. It is included for possible future application.
fontfamily	bch	The fontfamily of the text watermark.
fontsize	1cm	The fontsize of the text watermark.[2.4]
fontseries	b	The font series of the text watermark.[2.5]
graphicsoptions		Additional user-supplied key-value options for the command \includegraphics.[2.6]
height, textheight, text-height	\paperwidth	The height of text watermarks.
mark, textmark, text-mark	DRAFT	The default text watermark.
picscale, picture-scale	1	Scale of picture watermark.
picbb, picture-bb, pic-bounding-box	0 0 100 100	The bounding box (dvi mode) or viewport (pdf mode) of the picture watermark.
picfile, picture-file		The filename of picture watermark.[2.7]
picfileext, pic-file-ext	eps/pdf	The filename extension of the picture watermark.[2.8]
picheight, pic-height, picture-height		The height of picture watermarks.
picangle, picture-angle	0°	The orientation of picture watermarks.
picwidth, picture-width		The width of picture watermarks.
picontoptext, picture-on-top-text	true	Pictures should be placed on top of text watermarks if the two types of watermark occur on the same spot on a page.
picxpos, picture-xpos, picture-x-position	0	The horizontal coordinate of the picture watermarks.
picypos, picture-ypos, picture-y-position	0	The horizontal coordinate of the picture watermarks.
scale, textscale, text-scale	1	The scale of the text watermark.

Continued on next page

Continued from last page

Option	Default	Meaning
showpagecenter, show-page-center, showpapercenter	false	Boolean for requesting the indication of the center of the paper. See xwmgrids package for additional information.
textontoppic, text-on-top-pic, text-on-top-picture	false	Text watermarks should be placed on top of graphics watermarks if the two types of watermark occur on the same spot on a page. This reverses the boolean picontoptext.
width, textwidth, text-width	\paperheight	The width of text watermarks; doesn't apply to graphics watermarks.[2.9]
xpos, textxpos, text-xpos, text-x-position	0	These specify the horizontal coordinate of watermark (with reference to the center of paper, and not the text center).[2.10]
ypos, textypos, text-ypos, text-y-position	0	The vertical coordinate of the watermark.
\newwallpaper macro[2.11]		
align, textalign, text-align	center	The internal horizontal alignment of the watermark within a watermark box.[2.12]
boxalign, box-align	center (s)	The horizontal and vertical alignments of the watermark box.[2.13]
picheight, picture-height	.25\paperheight	The height of each cell of picture wallpaper.
picwidth, picture-width	.25\paperwidth	The width of each cell of picture wallpaper.
tilexsize, tile-xsize	.25\paperwidth	The width of each cell of tiled wallpaper.
tileysize, tile-ysize	.25\paperheight	The height of each cell of tiled wallpaper.
textheight, text-height	.25\paperheight	The height of each cell of text wallpaper.
textwidth, text-width	.25\paperwidth	The width of each cell of text wallpaper.
tilexoffset, tile-xoffset	0pt	The horizontal shift between the tiles of wallpaper.
tileyoffset, tile-yoffset	0pt	The vertical shift between the tiles of wallpaper.
tileno, tilenumber, tile-number, number-of-tiles	4	The maximum number of cells for tiled wallpapers if the parameters tilexsize and tileysize aren't specified or if the boolean squaretiles is true.[2.14]
squaretiles, square-tiles	true	The boolean that determines if the tiles should be rectangular or square.[2.15]
wpxoffset, wallpaper-xoffset	0pt	The horizontal offset of wallpaper from the margin or edge of paper.
wpyoffset, wallpaper-yoffset	0pt	The vertical offset of wallpaper from the margin or edge of paper.
\xwmminipage macro[2.16]		
depth	0ex	Additional depth of the boxed minipage.
height	.5\textheight	Additional height of the boxed minipage.
framebox, insertframe, insert-frame	true	The boolean that determines whether the box is framed or not.
framesep	3pt	\fboxsep of the boxed minipage.

Continued on next page

Continued from last page

Option	Default	Meaning
framerule	0.4pt	\fboxrule of the boxed minipage.
textwidth	\paperwidth	Width of the boxed minipage.
textcolor	black	Color of text inside boxed minipage.
framecolor	white	Color of frame of boxed minipage.
textalign, text-align	center	Alignment of the text inside the boxed minipage of the command \xwmminipage (expected values are center, left, right and justified).
\xwmcolorbox macro		
depth	0ex	Additional depth of the colorbox.
framebox, insertframe, insert-frame	true	The boolean that determines whether the box is framed or not.
framesep	3pt	\fboxsep of the colorbox.
framerule	0.4pt	\fboxrule of the colorbox.
height	0ex	Additional height of the colorbox.
innerframecolor, inner-frame-color	white	Color of the inner frame of colorbox.
outerframecolor, outer-frame-color	white	Color of the outer frame of colorbox.
textcolor	black	Color of text inside colorbox.
textalign, text-align	center	Alignment of the text inside the colorbox of \xwmcolorbox (expected values are center, left, right and justified).
width	\paperwidth	Width of the colorbox.
\makecolorbox macro		
The options of the macro \makecolorbox are the same as those of the related macro \xwmcolorbox. Please refer to subsection 7.3 for the syntaxes of these macros.		
\fancypagenos macro		
cfoot, center-footer, footer-center		Center footer.[2.17]
chead, center-header, header-center		Center header.
format	\fbox	The desired format of the fancy-page numbers.
coordunit, coord-unit, position-unit	mm	The units of the position/coordinates x and y.
fontfamily	cmss	The fontfamily of the fancy-page numbers.
fontsize	5cm	The fontsize of the fancy-page numbers.
fontseries	b	The font series of the fancy-page numbers.
footrule-height	.4pt	Foot rule height.
footrule-depth	.4pt	Foot rule depth.
footrule-width	\headwidth	Foot rule width.
footrule-sep, footrule-separation	2pt	Foot rule vertical separation (i.e., the vertical separation between the two lines.
footrule-color	cyan	Foot rule color.

Continued on next page

Continued from last page

Option	Default	Meaning
hfoffset-left	.5	Left \hfoffset (see fancyhdr for the meaning of \hfoffset).
hfoffset-right	.5	Right \hfoffset (see fancyhdr).
headrule-height	.4pt	Head rule height.
headrule-depth	.4pt	Head rule depth.
headrule-width	\headwidth	Head rule width.
headrule-sep, headrule-separation	2pt	Head rule separation (i.e., the vertical separation between the two lines.
headrule-color	blue	Head rule color.
lfoot, left-footer, footer-left		Left footer.
lhead, left-header, header-left		Left header.
style, printstyle	\thepage	The presentation style of the fancy-page numbers.
rfoot, right-footer, footer-right		Right footer.
rhead, right-header, header-right		Right header.
showpagenos, show-pageno, show-pagenos, show-page-numbers	true	Boolean that determines if fancy-page numbers should be shown.
showonpageone, show-on-page-one	false	Boolean that determines if fancy-page numbers should be shown on the first page of the document.
showheadrule, show-headrule	true	Boolean that determines if the headrule should be shown.
showfootrule, show-footrule	true	Boolean that determines if the footrule should be shown.
sendtoback	true	Boolean that determines if the fancy-page numbers should be placed in the background or foreground.
align, textalign, text-align	center	The alignment of the text in the fancy-page number box (if it is boxed).
boxalign, box-align	center	The alignment of the fancy-page number box (if it is boxed).
color, textcolor, text-color	blue	The color of the fancy-page numbers.
scale, textscale, text-scale	2.5	The scale of the fancy-page numbers.
angle, textangle, text-angle	0	The orientation of the fancy-page numbers.
width, textwidth, text-width	0ex	Width of the fancy-page numbers.
height, textheight, text-height	0ex	Height of the fancy-page numbers.

Continued on next page

Option	Default	Meaning
Continued from last page		
xpos, textxpos, x-position	0	The horizontal coordinates of the fancy-page numbers (from page center).
ypos, textypos, y-position	0	The vertical coordinates of the fancy-page numbers (from page center).

Table 2 notes

[2.1] The key can assume only one of the values center, left, right and justified. Any other text alignment value is inadmissible and thus rejected by the package with a fatal error.

[2.2] The options allpages, evenpages, oddpages are boolean keys, while page, pages and pagex require values, e. g., page=x, pages=x-y and pagex={x,y,z}. If the package is loaded and none of these options is passed to it, but the option printwatermark is true, the default watermark (DRAFT) will be printed on the first page of the document (but only in draft mode) and a warning message logged in the transcript file. The key pages expects a page range with the pages separated by a hyphen, while pagex expects a list of pages that will receive the watermarks.

[2.3] Both text and picture watermarks are normally put in boxes for manipulation before being typeset. This key refers to the horizontal and vertical alignments of the box. The key can assume one of the values t-l, t-r, b-l, b-r, s, or top-left, top-right, bottom-left, bottom-right, center, justified. Elements of these two sets can't be combined. When submitted as key values, the elements of both sets are not to be separated by commas or spaces but by hyphens (e. g., one of t-l, t-r, b-l, b-r, s, or one of top-left, top-right, bottom-left, bottom-right, center, justified).

[2.4] If for some reason you need other fontsizes (e. g., 10pt, 11pt or 12pt for printing text watermarks in \normalfont), you will need to submit them as values of fontsize.

[2.5] If you need normal document text, put fontseries=m, which implies medium weight and width.

[2.6] The command \includegraphics is used for all graphics inclusion tasks. Users can directly pass values to admissible options of the command \includegraphics. See subsection 5.1.

[2.7] This, with its full path, must be submitted when including picture watermark.

[2.8] Valid extensions are eps, pdf, png, jpeg, mps; the latter four may be used in the case of PDFTEX. The file extension should be passed without the dot. If the option is not passed to package, xwatermark selects eps (in dvi mode) or pdf (in PDFTEX mode).

[2.9] For some design reasons we set \paperheight as the default value of mark's width instead of mark's height.

[2.10] There is no need to attach a unit to xpos or ypos; if the user does, the unit will be used in place of coordunit.

[2.11] Some of the \newwatermark keys can also be called when inserting wallpapers. If you aren't sure if a key is applicable, don't worry: just try it. The package collects all inadmissible keys together and prints them in the transcript file on each run. Normally, the user is alerted by an error message. Indeed if you receive any failed compilation, you should first suspect that the failure is due to the use of an invalid key. The next keys apply specifically to wallpaper watermarks.

[2.12] The key can assume only one of the values center, left, right and justified.

[2.13] The key can assume one of the values t-l, t-r, b-l, b-r, s, or top-left, top-right, bottom-left, bottom-right, center, justified. When submitted as key values, the elements are not to be separated by commas or spaces but by hyphens (e. g., one of t-l, t-r, b-l, b-r, s, or one of top-left, top-right, bottom-left, bottom-right, center, justified).

[2.14] If tilexsize and tileysize are given and squaretiles is false, tileno is ignored.

[2.15] Sometimes choosing squaretiles (i. e., setting it true) can yield an undesired outcome if textalign and boxalign haven't been properly selected. In such a situation, the user is advised to first experiment with setting squaretiles to false and/or textalign and boxalign to center.

[2.16] The \xwmminipage command can take verbatim material and provides a ⟨key⟩=⟨value⟩ alternative to the \boxedminipage command of the boxedminipage package.

[2.17] All headers and footers are as in fancyhdr package.

14   VERSION HISTORY

The following change history highlights significant changes that affect user utilities and interfaces; changes of technical nature are not documented in this section. The star sign ($\star$) on the right-hand side of the following lists means the subject features in the package but is not reflected anywhere in this user guide.

Version 1.5.2   [2011/10/20]

To match changes in `ltxkeys` package   . . . . $\star$

Version 1.5.1   [2011/07/20]

Following user request, two new keys were introduced for the macro `\fancypagenos` .   section 8

Version 1.5.0   [2011/07/10]

Introduced the `ltxkeys` package, a highly robust and optimized module for general creation and management of keys . . . . $\star$

Provisions for placing more than one watermark on the same page.   . . . .   section 3

Introduced wallpaper functionalities . . . .   section 4

Adaptable and flexible fancy page numbers . . . .   section 8

INDEX

Index numbers refer to page numbers.