The **skrapport** document class*†

Simon Sigurdhsson ⌈sigurdhsson@gmail.com⌋
Version 0.12e

Abstract A document class intended for simple documents *e.g.* reports handed in to courses and such. It is small, straightforward and heavily inspired by the PracTeX Journal style.

Contents

---

*Available on `http://www.ctan.org/pkg/skrapport`.
†Development version available on `https://github.com/urdh/skrapport`.

1 Documentation

The skrapport document class aims to make typesetting simple but stylish documents (mostly reports) as effortless as possible. It does this by mostly reimplementing the default article class in LaTeX3, while making modifications to both form and function along the way.

Because it is reimplemented in LaTeX3, it may be incompatible with any number of packages that patch or otherwise modify internals of article or other document classes. For commonly used packages (especially those used frequently by the author), this shouldn't be a problem. The author gladly accepts reports of any such issues at the project issue tracker — see 'Known issues' on on page 14.

1.1 Options

As with other document classes, the class is loaded, possibly with options, by issuing **\documentclass**[⟨*options*⟩]{skrapport}. The class has a number of options controlling both form and function, by *e.g.* setting the font size, selecting a font stack, setting the section title style, and so on.

1.1.1 Layout

Two options controlling the overall layout of the document are provided. Collectively they control the paper size and text layout of the document.

paper   a4, a5   (a4)

The paper option controls the paper size of the document. Internally, this is set by the typearea package, so in theory many more paper sizes could be available, but the current options cover most useful documents.

twocolumn   The twocolumn option sets up a two-column mode. This is not provided by internal LaTeX mechanics as in the original article class, but instead by patching environments and macros and using a combination of the multicol and grid packages. In theory, this means that baselines

of adjacent columns should be aligned, and that three- or four-column modes are possible in the future (but that'd be ridiculous).

1.1.2 Style

A couple of options to control the style of the document are provided. Two of them, `leqno` and `fleqn`, are mainly provided for compatibility with the article class.

`leqno`
This option makes display math environments typeset their labels on the left-hand side of the formula instead of the right-hand side.

`fleqn`
This option makes display math environments left-align the entire formula as opposed to centering it.

`indent` `true, false` *(false)*
The `indent` option enables or disables the indentation of paragraphs, with the default being not to indent anything. The default behaviour thus is similar to that obtained using the parskip package with the article package.

`titles` `rm, it, bf, sf` *(bf)*
Section titles (and a few other elements) are controlled by this option in that they are typeset either using the regular roman font, the boldfaced roman font or the sans serif font. For historical reasons the default is a boldfaced roman font, but the sans serif option is very handsome.

`hanging-titles` `true, false` *(false)*
This options allow sections to be set as 'hanging' titles, *i.e.* with the section number in the margin.

`color` *⟨color theme⟩* *(default)*
This option tells the class to activate color theme support and optionally load a color theme. Several color themes are available (as detailed by 'Color themes' on page 13), and the special value `false` disables color support entirely (which only means that the xcolor package isn't loaded, and that `\colortheme` remains undefined).

1.1.3 Fonts

Only two options control the font setup of the document class. The class provides the same point sizes as article, but also provides a large number

of font stacks to choose from.

`ptsize`   `10pt, 11pt, 12pt`   (11pt)
The document class provides the same three point sizes as the article class. There is room for expansion, but there really shouldn't be any reason to use other point sizes.

`font`   `none, kpfonts, lmodern, palatino, minion, skdoc, word`   (kpfonts, word)
Several different font stacks, detailed by table 1, are provided. Most of these work with pdfLaTeX, but there are two special font stacks: `none` and `word`. The former, predictably, loads no fonts whatsoever leaving the document with Computer Modern fonts. This is useful if the fonts are replaced later anyway, such as when using XƎLaTeX.

The `word` font stack, however, does load a few fonts. The fonts must be present and installed on the system as OTF or TTF fonts, and the font stack requires either XƎLaTeX or LuaLaTeX. Both `word` and `none` load the fontspec package, assuming the document is compiled using XƎLaTeX or LuaLaTeX.

(***Note:*** *The* `skdoc` *font stack doesn't have a math font.*)

Warning: Currently, fontspec is always loaded by XƎLaTeX regardless of options due to the use of polyglossia. This means that you *have* to use OpenType fonts in your document, otherwise *only* Latin Modern Roman (*i.e.* no sans-serif or monospace fonts) will be available. Therefore, the `word` font stack is the default when using XƎLaTeX. LuaLaTeX is not affected by this.

1.1.4 Functionality

The final three options affect functionality in one way or another. Since skrapport was originally designed for reports written in either swedish or english, the class always loads either babel or polyglossia with either of these languages. Additionally, the class may load the skmath package if desirable.

`nomath`   `true, false`   (false)
When `false`, the skmath package is loaded, providing improvements to the math functionality of amsmath and friends.

Table 1: *Font stacks provided by skrapport.*

Font stack	Serif font	Math font	Sans serif font	Monospace font
kpfonts	Kp-Fonts	Kp-Fonts	Kp-Fonts	Kp-Fonts
lmodern	Latin Modern	Latin Modern	Latin Modern	Source Code Pro
palatino	TeX Gyre	PX	Arev	Source Code Pro
minion	Minion Pro	Minion Pro	Myriad Pro	Source Code Pro
skdoc	PT Serif	—	Open Sans	Source Code Pro
word	Cambria	Cambria Math	Calibri	Consolas

lang   en, sv, de   (sv)
This option specifies what main language babel or polyglossia set up with. English (or swedish, for lang=en) is loaded as well, for use in constructs that allow for a second language.

draft   true, false   (false)
The draft option, much like in article, enables \overfullrules and possibly similar functionality in loaded packages (if they react to the global draft option).

1.2 User-level commands and environments

The general idea behind the document class is to provide most (if not all) of the macros provided by the standard LaTeX 2$_\varepsilon$ classes, as well as additional macros to simplify and beautify the documents produced. As such, most of the documentation that follows details macros that are present in the standard document classes. Some of them have changed functionality or semantics, so at least a skim through this section is recommended.

1.2.1 The front page

The front page is the part of the document that has seen the most changes in skrapport compared to article. In addition to the new, PracTeX-inspired layout, there are a few additional pieces of information in it. Also, the \author macro has been dramatically improved.

\date   {⟨*ISO8601 date*⟩}

The \date macro now internally employs isodate to typeset the date of the document. This means that the input must be either the string 'today', the token \today or a date as defined by ISO8601[1]. The mechanisms provided by isodate can be used to affect the output format.

\author   [⟨*email*⟩] {⟨*name*⟩}

In contrast with the article package, the \author macro should no longer be used to typeset several author names at once. Instead, one \author command is to be issued for every author, optionally providing a corresponding email. These are then combined, in the order they are given, to form a list of authors and a corresponding list of email adresses.

\regarding   {⟨*topic*⟩}

This macro defines a topic or other short message detailing the purpose of the document. It is typeset along with the date in the upper left corner of the title page.

\license   {⟨*license name*⟩}

The \license macro defines a short license name to be typeset in the lower right corner of the title page. This mechanism could of course be used to typeset an institution name or similar as well. It is only typeset when also using the titlepage environment described below.

\title   {⟨*document title*⟩}

The \title macro, while not defined by skrapport, is relevant to describe. It simply sets the title of the document, as displayed by \maketitle.

\maketitle   [hide={date,regarding,email},nopdfinfo]

As in article, the \maketitle macro typesets the information provided by \title, \author and friends to form a stylish front page. When

---

[1] International Organization for Standardization, Technical Committee 154 2004.

combined with abstract, titlepage and/or \tableofcontents, you get a very good-looking preamble with almost no effort.

The optional argument is a key-value list with two valid entries, hide and nopdfinfo. The hide entry accepts a comma-separated list in which the values date, regarding and email are interpreted as instructions to hide the corresponding entry from the front page. The nopdfinfo key disables the generation of PDF information by hyperref.

\begin{**abstract**}
⟨*abstract text*⟩
\end{**abstract**}

The abstract environment defines an abstract, which is typeset in a block with the \abstractname word next to it (see *e.g.* the title page of this manual for an example).

\begin{**titlepage**}
⟨*title page contents*⟩
\end{**titlepage**}

Normally, \maketitle doesn't reserve its own page. By enclosing \maketitle (and abstract if appropriate) in the titlepage environment, the contents are typeset on their own page, without page numbering and with the \license text in the lower right corner (if applicable).

1.2.2 Sectioning

\section	⋆[⟨*short title*⟩]{⟨**title**⟩}
\subsection	⋆[⟨*short title*⟩]{⟨**title**⟩}
\subsubsection	⋆[⟨*short title*⟩]{⟨**title**⟩}
\paragraph	⋆[⟨*short title*⟩]{⟨**title**⟩}
\subparagraph	⋆[⟨*short title*⟩]{⟨**title**⟩}

The sectioning macros are superficially very similar to those provided by article, but have been completely reimplemented in expl3 code. Additionally, the style of these sectioning macros, as detailed by the titles option documentation above, can be changed.

All of them have both arguments and behaviour in common, only differing in style. The starred versions are unnumbered, but still accept the opional short title (simply discarding it). The optional short title

is used in the table of contents. The `secnumdepth` counter limits the depth of section numbering.

The first three macros, being section titles, are typeset as actual titles on their own line with appropriate spacing above and below. The paragraph macros instead typeset run-in titles.

1.2.3 Macros and environments from article

Aside from the `itemize` and `enumerate` envionments and the font selection macros, which are carried without modification from LaTeX 2_ε, a couple of environments and macros are defined.

The old font macros, whose use has been discouraged for a long time, are now deprecated. Instead of functioning correctly, they will emit an error and require user input. Use the `\text??`/`\??family` macros instead. For emphasis, use the `\emph` macro.

\emph {⟨*text*⟩}

Emphasized text will be typeset in italic, or bold italic if the macro is used in a context where italic is already used (such as inside another `\emph` invokation).

begin{**description**}

\item[⟨*item*⟩] {⟨*description*⟩}

\end{**description**}

The `description` environment behaves as expected, typesetting a list of descriptions as in the article class.

\begin{**quote**}
⟨*short quote*⟩
\end{**quote**}

Intended for short quotes, the `quote` environment simply typesets a centered block of italic text.

\begin{**quotation**}
⟨*long quote*⟩
\end{**quotation**}

Longer quoted passages are typeset using the `quotation` environment. This is simply a `quote` environment with additional spacing above and below.

\begin{**verse**}
⟨*pretentious poetry*⟩
\end{**verse**}

The `verse` environment is intended for poetry and other text where line breaks are critical. Use `\\` to break lines.

\appendix

This macro signals the end of the main matter and the start of the appendix. In essence, it resets the section numbering counter and changes the section numbering to the upper-case alphabetic sequence.

1.2.4 Floats

Both the `figure` and `table` float environments accept an optional positioning argument. The default positioning is `tp`. Both environments also have starred variants, which do nothing in one-column mode while typesetting the figure across both columns in two-column mode. As usual, `\centering`, `\caption` and `\label` should be used inside the floats.

\begin{**figure**} `[position]`
⟨*content*⟩ This float environment is intended for figures. The most common con-
\end{**figure**} tents are `\includegraphics` statements or `tikzpicture` environ-
ments.

\begin{**table**} `[position]`
⟨*content*⟩ A float intended for tables. Probably contains `tabular`s.
\end{**table**}
\begin{**figcenter**}
\end{**figcenter**} This environment is useful for wide figures and tables. It typesets its contents centered horizonally, but allows the content to extend into the margin. The content is set in a horizontal coffin.

1.2.5 Table of contents

\tableofcontents

The table of contents are typeset using this macro. The `tocdepth` counter limits the depth of the table of contents, but for stylistic reasons values higher than 3 are unsupported.

1.2.6 Miscellaneous

\comment ⋆{⟨*comment*⟩}
    \note ⋆{⟨*comment*⟩}
    \com ⋆{⟨*comment*⟩}

These macros, the two latter being aliases of the first one, typeset an author's comment in the document. The starred variants typeset the comment inline with a red background, while the unstarred variant typesets the comment in a \marginpar.

\eg
\ie
\etc
\cf
\viz

These macro print the abbreviation of the latin phrases *exempli gratia*, *id est*, *et cetera*, *confer* and *videlicet*, respectively. The macros peek ahead to find punctuation marks and spaces, so they should behave correctly regardless of usage (assuming they're used in running text and uncomplicated settings). In languages other than english, appropriate translations are made if applicable.

\dash

This macro prints an em-dash surrounded by thin spaces, as discussed by Flynn (2006, p. 8). When \DeclareUnicodeCharacter is available, the real em dash uses this definition.

1.2.7 Color theme support

\colortheme {⟨*theme*⟩}

The \colortheme macro, which is only available when the color option is true, applies a color theme to the document. For a list of available color themes, see 'Color themes' on page 13.

1.2.8 Font size macros

The font size macros, expectedly, set the size of the text. They do not take arguments, instead affecting all subsequent text of the current TEX group, so use braces to provide and limit context. Also note that unlike article, these macros are *all* available, regardless of point size option.

\tiny

Typesets tiny text.

\scriptsize

Typesets script-size text.

\footnotesize

Typesets footnote-sized text.

\small

Typesets small text.

\normalsize

Typesets normal text.

\large

Typesets large text.

\Large

Typesets slightly larger text.

\LARGE

Typesets even larger text.

\huge

Typesets huge text.

\Huge

Typesets really huge text.

1.3 Color themes

\colortheme {⟨*theme*⟩}

If the package is loaded with the color option, changing the color theme is possible using \colortheme, which loads an appropriate package. At the moment, four color themes are available.

default
The default theme is fairly conservative, only coloring hyperef links with more readable, slightly darker colors than the standard ones. It should print well even on non-color printers.

unscathed
The unscathed theme is based on a palette with the same name on COLOURlovers[2], and applies a dark brown color to emphasized text, a rusty color to links, a darker rust color to titles and a lighter brown to quotes.

cruelwater
The cruelwater theme is also based on a palette from COLOURlovers[3], and applies a dark blue color to bold text and captions, a slightly less dark blue to titles and emphasized text, a light gray color to small print and a darker gray to quotes.

violet
The violet theme, like unscathed and cruelwater, is based on a COLOURlovers palette[4]. It colors all links bright purple, applies a dark puple color to titles, bold text and captions, a grayish purple to small print, a dark brown color to quotes and a pastel violet color to emphasized text.

---

[2]http://www.colourlovers.com/palette/1440498/unscathed
[3]http://www.colourlovers.com/palette/126030/Cruel_Water_at_
  Night
[4]http://www.colourlovers.com/palette/1831303/Violet_White_
  Bedrm

skdoc   The skdoc theme is loosely based on the skdoc document class, with which this documentation is typeset.

2 Known issues

A list of current issues is available in the Github repository of this package[5], but as of the release of v0.12e, there is one known issue:

– If a `\subsubsection` is the last item of the Table of Contents, it will not be indented properly.

If you discover any bugs in this package, please report them to the issue tracker in the skrapport Github repository.

---

[5] `https://github.com/urdh/skrapport/issues`

3 Installation

The easiest way to install this package is using the package manager provided by your LaTeX installation if such a program is available. Failing that, provided you have obtained the package source (`skrapport.tex` and `Makefile`) from either CTAN or Github, running `make install` inside the source directory works well. This will extract the documentation and code from `skrapport.tex`, install all files into the TDS tree at TEXMFHOME and run `mktexlsr`.

If you want to extract code and documentation without installing the package, run `make all` instead. If you insist on not using `make`, remember that packages distributed using skdoc must be extracted using `pdflatex`, *not* `tex` or `latex`.

4 Changes

v0.01

General: Initial version.

v0.03

General: Removed `\rd` and `\id`.

v0.04

General: Added microtype package.

v0.05

General: Improved documentation.

v0.06

General: Corrected cheksum, further improved documentation.

v0.07

General: Various bugfixes, X⿰LaTeX compatibility, better float settings, quote style fix, `intlimits` option to amsmath.

v0.07a

General: Fixed kpfonts issues.

v0.09

General: Introduced kvoptions, fixed abstract in two-column mode.

v0.10

General: Include skmath if exists and wanted. Gobble

optional arguments to `figure` and `table` in two-column mode.

The `word` font stack is now the default font stack on XƎLaTeX.

5 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the page were the implementation of the corresponding entry is discussed. Numbers in roman refer to other mentions of the entry.

6 Bibliography

Flynn, Peter (2006). 'Rolling your own Document Class: Using LaTeX to keep away from the Dark Side'. In: *The PracTeX Journal* 4. URL: http://tug.org/pracjourn/2006-4/flynn/flynn.pdf.

International Organization for Standardization, Technical Committee 154 (2004). *ISO 8601:2004. Data elements and interchange formats – Information interchange – Representation of dates and times*.