

The phfnote package¹

Philippe Faist *philippe.faist@bluewin.ch*

August 15, 2016

¹This document corresponds to *phfnote v1.0*, dated 2016/08/15. It is part of the *phfqitlx* package suite, see <https://github.com/phfaist/phfqitlx>.

phfnote—A handy \LaTeX class for typesetting short notes and medium-length reports, full of goodies to make it look just right.

1	Introduction	2
2	Basic Usage	3
2.1	Loading the Package	3
2.2	Presets	3
3	Features	5
3.1	Title Formatting	6
3.1.1	Title Styles	6
3.1.2	Customizing the style of the <code>default</code> and <code>small</code> title styles	7
3.1.3	Title notes: <code>\thanks</code> and <code>\thanksmark</code>	8
3.2	Abstract	9
3.3	Table of Contents	10
3.4	Predefined Package Sets	11
3.5	Page Geometry	11
3.6	Section Headers Styling	13
3.7	Appearance of Paragraphs	15
3.8	Adjusting Spacing of Lines and Words	16
3.9	Adjustments for Fonts	16
3.10	Footnote Figure Style	16
3.11	Hyperref Loading	17
3.12	Bibliography Definitions	18
3.13	URL Styles	18
3.14	A <code>\notesmaller</code> Command	19
3.15	Tools Mostly for Hackers	19
4	Summary of Package Options	19
5	Implementation	21
5.1	Internal Generic Code	21
5.2	Title Styling	22
5.2.1	First, some common simple definitions for our different styles	22
5.2.2	Implementation of <code>\thanks</code> and <code>\thanksmark</code>	23

5.2.3	Title Styles Definition	25
5.2.4	Plugging into <code>\maketitle</code>	27
5.3	Abstract	28
5.4	Page Geometry Settings	30
5.5	Text, Paragraph and Line Spacing	32
5.6	Section Styling	33
5.7	\LaTeX Package Sets	35
5.8	Hyperref Support and Hyperlinks	37
5.9	Cosmetic Font Definitions	39
5.10	Bibliography Stuff	39
5.11	Better Footnote Style	40
5.12	Other Stand-Alone Definitions and Helpers	41
5.12.1	A <code>\notesmaller</code> command	41
5.12.2	Customized, “Inline,” Table of Contents	41
5.12.3	URL Styles	42
5.12.4	Utility to Add TOC Entry For Starred Section	43
5.12.5	Hack to save & restore a set of commands	43
5.12.6	A utility for verbatim stuff in arguments of other macros	45
5.13	Handle Package Options	45
5.13.1	Define and Parse Package Options	45
5.13.2	Define Global Presets	47
5.13.3	Finally, Process and Execute the Package Options	56
	Change History	57
	Index	57

■ 1 Introduction

Have you ever thought, “let me write up these short notes using \LaTeX ,” but then disliked the default style of the `article` class? Have you ever asked yourself why half the page should be taken up by the title? Yes? Then welcome to `phfnote`.

The package `phfnote` provides basic formatting for short documents, such as notes on a specific topic, short documentation, or quick memos. It aims to cover all basic needs for such purposes: include a standard set of relevant packages, a nice title which doesn’t take up too much space, better page margin sizes, and some basic styling to make the note look nicer. At the same time, it is highly configurable so that nothing is really unchangeable. And all overridden features can be restored individually to their class-provided defaults.

This package has been designed to work optimally along with the `article` document class, but in principle any relatively standard \LaTeX class should work.

Notes can be typeset in two-column mode with the `twocolumn` option of for example the `article` class. Settings such as the page margins and font goodies are automatically adapted to look best according to the standard document font size (10pt, 11pt, or 12pt).

Be aware that this package is not meant as a full-fledged formatting class for complicated articles. For that, you should use a specialized class such as `REVTeX`.¹

In the following, we detail individual features of this class, and explain how to activate, deactivate, and customize them.

■ 2 Basic Usage

2.1 Loading the Package

You can get started with the minimal template:

```
\documentclass[11pt,a4paper]{article}
\usepackage{phfnote}

\begin{document}
\title{Title of my notes}
\author{Me}
\date{\today}
\maketitle

...

\end{document}
```

The package `phfnote` introduces its default note formatting style, with a more compact title, and some formatting adjustments in the text and section headings.

2.2 Presets

There are a number of package options which can be provided to activate, deactivate or adjust the formatting. The most straightforward way of changing the formatting is to use *presets*.

Presets are processed immediately when given in the package option list, meaning that their position in the list is meaningful. For example, the option list

```
\usepackage[title=small,preset=article,par=skip]{phfnote}
```

¹See <https://journals.aps.org/revtex>

will set `title=small` only if it is not overridden by the `article` preset, but will enforce `par=skip` in any case. You may in theory load several presets, e.g. `preset=sfnote,preset=article`, but this is essentially useless since presets tend to set a wide range of settings such that in any case the last preset specified is effectively applied.

First, there is a set of presets which are different alternative “note” styles. All the following define the note to have spacing between paragraphs and no first line indentation, use the default note title style, and use a wider page geometry.

`preset=sfnote`

Format the note in L^AT_EX’ sans-serif “Computer Modern Bright” font. This is a nice, light, font for short notes, but I find it more difficult to read at smaller font sizes or in longer paragraphs.

`preset=sfssnote`

Format the note in L^AT_EX’ default sans-serif font. A very nice sans serif font. It might look heavy though, depending on your taste.

`preset=opensansnote`

Format the note in Open Sans font (using the ‘opensans’ package with some default options). A very beautiful and readable sans serif font.

`preset=utopianote`

Format the note in Utopia font (by using the `fourier` package). Perfect to my taste for documenting code for example, but I find it a bit heavy for scientific documents.

`preset=mnmynote`

Format the note in Minion Pro font, with sans serif text formatted with the Myriad Pro font (professional fonts by Adobe which can be used in L^AT_EX with the `MinionPro` and `MyriadPro` packages²). These beautiful fonts can be used for any purpose.

Based essentially on `utopianote`, the preset `pkgdoc` sets up the document to look nice for a L^AT_EX package documentation. The preset `xpkgdoc` adds additional definitions to aid in documenting L^AT_EX packages on top of `pkgdoc`.

`preset=pkgdoc`

Basic formatting and settings for documenting L^AT_EX packages. This preset was used for the current document.

`preset=xpkgdoc`

Same as `preset=pkgdoc`, but in addition a set of useful commands are

²See <https://github.com/sebschub/FontPro>; the fonts themselves ship with some Adobe products

also provided, the `tcolorbox` package is loaded along with some default boxes. Also some commands are patched to achieve some fixes. This preset is used for the documentation of packages in the `phfqitlx` package suite. (For details see the implementation of `\phfnote@preset@xpkgdoc` below.)

The following preset makes the document look more like an article. There are some slight minor differences with respect to the default `article` class' title in the choice of formatting the title and text.

```
preset=article
```

Sets a more title style closer to `article`'s default title style (but slightly more compact) and sets paragraphs to indent with no skip.

The last preset, `reset`, guarantees that including this package is non-invasive, meaning that only new \LaTeX macros are made available without altering any appearance. This is useful if you want to use a small feature provided by this package, but you already have all the page geometry, title, etc. set up and want to make sure those aren't touched.

```
preset=reset
```

Deactivates all features of this package by default. Individual settings can still later be switched on via specific package options. Use this to activate only a specific set of features: `[preset=reset, . . .]` will ensure that only the additional given features are set.

This is safer than deactivating individually all other features, because in the future we may add new features which may be on by default. In this case, the preset `reset` will guarantee all features to be deactivated.

■ 3 Features

This package provides a large collection of small features, which, put all together, make the document look nicer (hopefully). Let's go through these features, one by one.

Note also that some features provided in the presets, such changing the document font, are not provided as individual features here. This is because they may be set and customized directly using few lines of \LaTeX code or directly by including an external package. In those cases, you may have a look at the preset's definition for inspiration (see [subsubsection 5.13.2](#)).

For a summary of package options, see [section 4](#).

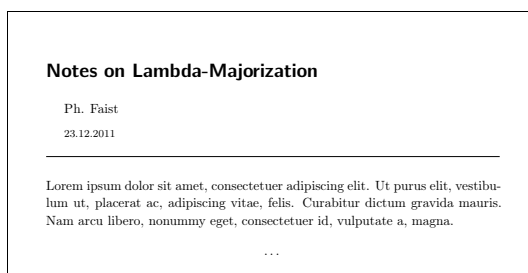
3.1 Title Formatting

3.1.1 Title Styles

The phfnote package allows a set of alternative title styles. By default, the default title style is used. You may change this setting with the `title=...` package option.

```
title=default
```

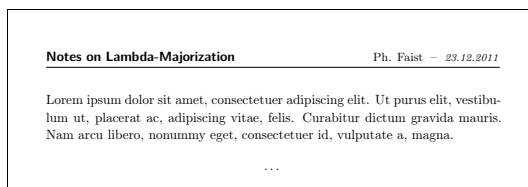
The default title style displays the title in large bold sans serif font, left-aligned. Below the title appears the information about author and date, indented, followed by a horizontal rule. It looks like this:



As you can see, it saves more space on the page compared to the default article title.

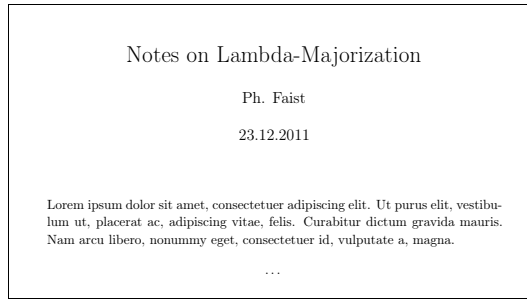
```
title=small
```

A smaller title style which displays all the relevant information on a single line. This is useful for when even the default title style appears too large. It looks like this:



```
title=article
```

Mimics the default title style from the `article` class, but saves a little more space. It looks like this:



`title=`

An empty argument to `title` instructs `phfnote` not to override any title definition, thus preserving the default class title style.

Beware that some other title goodies, such as our more advanced `\thanks` notes, or spacing adjustments for the abstract, will probably not work.

3.1.2 Customizing the style of the default and small title styles

You may customize the appearance of the default and small title styles by overriding some macros.

`\notetitlefont` The macros `\notetitlefont`, `\notetitleauthorfont`, and
`\notetitleauthorfont` `\notetitledatefont` set the default main font title, author text and
`\notetitledatefont` date text. You may override these settings with, for instance:

```
\renewcommand{\notetitlefont}{\sffamily\bfseries}
\renewcommand{\notetitleauthorfont}{\itshape}
\renewcommand{\notetitledatefont}{\footnotesize}
```

`\notetitlebelowspace` The spacing of the title may be adjusted with the macros
`\notetitlebelowspace` `\notetitlebelowspace` and `\notetitletopspace`. Override these
`\notetitletopspace` with e.g.:

```
\renewcommand{\notetitlebelowspace}{4mm}
\renewcommand{\notetitletopspace}{-1.2cm}
```

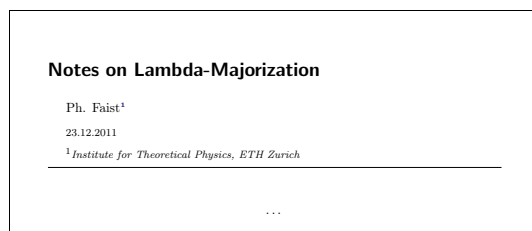
`\notetitlehrule` Finally, you may override the command `\notetitlehrule` which draws the
rule below the title:

```
\renewcommand{\notetitlehrule}{\hrule height 0.8pt}
```

3.1.3 Title notes: `\thanks` and `\thanksmark`

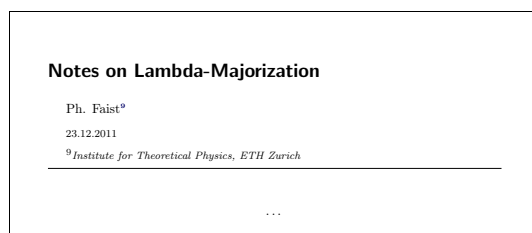
`\thanks` Notes in the title can be introduced with the `\thanks` macro. You may use this to specify an e-mail address, an affiliation, or any other more specific information. `\thanks` may appear in all three title, authors and date.

The appearance of this additional information depends on the title style. In the default note title style, such thanks-notes appear directly below the title. For example, with `\author{Ph. Faist\thanks{\itshape Institute for Theoretical Physics, ETH Zurich}}`, you get:



whereas with the other styles, this information is typeset as regular footnotes.

`\thanks[N]` You may specify an optional argument to `\thanks`, forcing the footnote to a specific number (it must be a number). For example, with `\author{Ph. Faist\thanks[9]{\itshape Institute for Theoretical Physics, ETH Zurich}}`, you get:



`\thanksmark` `\thanksmark[N]` works with `\thanks` as `\footnotemark` works with `\footnote`. It just displays the given number as a footnote mark. In this way, you can have for example several shared affiliations:



the author code was:


```
\author{First Author\thanks[1]{\itshape Institute ABC},
        Second Author\thanks[2]{\itshape Somewhere else},
        and Third Author\thanksmark[1]}
```

Unfortunately, you still have to provide the numbering manually. On the other hand, this package is not meant to replace REVTeX, so if you're writing a complicated article with many authors and affiliations, you probably shouldn't be using phfnote in the first place.

WARNING

The optional argument to `\thanks`, as well as the command `\thanksmark`, are not made available if you don't use one of `\phfnote`'s title styles.

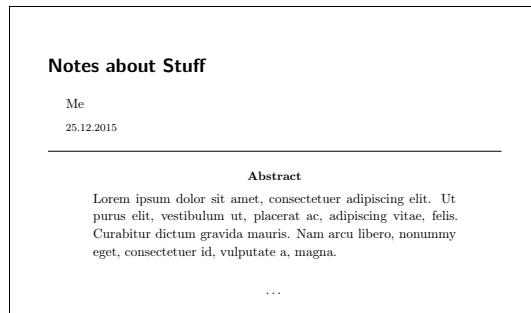
This behavior is such as to prevent interference with more advanced class mechanisms, such as REVTeX's.

3.2 Abstract

`abstract` The `abstract` environment renders indented text aimed to provide a short summary of the document. We might use, for example, the following code:

```
\begin{abstract}
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus
elit, vestibulum ut, placerat ac, adipiscing vitae, felis.
Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget,
consectetur id, vulputate a, magna.
\end{abstract}
```

which would look like this:



The `abstract` environment should be given *after* the `\maketitle` command. (In contrast to, e.g., REVTeX.)

You may customize the appearance of the abstract via a list of attributes given as argument to a package option. When you combine arguments, make sure to put them in a braced group: `[abstract={wide,noname,it}]`.

`abstract={wide,...}`

The abstract should not be indented, and should instead be aligned to the rest of the text.

`abstract={narrow,...}`

The abstract should be indented narrower than by default.

`abstract={noname,...}`

The title “Abstract.” above the text will not be typeset. The abstract text is typeset directly instead.

`abstract={original,...}`

Revert to the class’ default implementation of the abstract environment. The class’ implementation is restored and no longer tampered with.

`abstract={small,...}`

Use a smaller font for the abstract text (`\small` font).

`abstract={compact,...}`

Reduce spacing before and after the abstract. If the abstract is short, this might look slightly better.

`abstract={it,...}`

Typeset the abstract text using an italic typeface.

The abstract environment’s appearance can be customized more finely by redefining some macros. (In fact, this is what the package options `abstract=...` actually do.) The font used for the text of the abstract is set by `\noteabstracttextfont`. This macro should expand to font selection commands, such as `\itshape`, `\bfseries`, `\small`, etc. The title of the abstract (the word “Abstract.”) is typeset in the font set by `\noteabstractnamefont`. The width of the whole abstract text is determined by `\noteabstracttextwidth`. Observe that `\noteabstracttextwidth` is a macro, and not a proper length, so that it can determine more dynamically the length. The spacing below (`\noteabstractafterspacing`) and above (`\noteabstractbeforespacing`) the abstract can further be specified, also as macros.

3.3 Table of Contents

`\inlinetoc` The package `phfnote` also provides a table of contents typeset with reduced

spacing to be more compact, and with horizontal rules before and after. You can insert the table of contents with the command `\inlinetoc`. It looks like this:

Notes about Stuff	
Me	
25.12.2015	
<hr/>	
Abstract	
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.	
<hr/>	
1	2
Introduction	
2	3
Basic Usage	
2.1 Loading the Package	3
2.2 Presets	3
<hr/>	
. . .	

3.4 Predefined Package Sets

The `phfnote` package also provides sets of standard \LaTeX packages to load. You may choose between a varying degree of “richness” of packages included.

`pkgset=none`

Do not include any package set.

`pkgset=minimal`

Include some basic minimal set useful for scientific notes: the $\mathcal{A}\mathcal{M}\mathcal{S}$ packages `amsmath`, `amssymb`, `amsfonts`, and `amsthm`. The `xcolor` package is also loaded.

`pkgset=rich`

Include a fair amount of packages which may be useful. On top of the minimal package set, this set includes the packages `enumitem`, `graphicx`, `microtype`, `caption`, `setspace`, as well as `inputenc` with the `utf8` option and `fontenc` with the `T1` option.

This package set is loaded by default.

`pkgset=extended`

Additionally, include packages `float`, `verbdef`, `csquotes`, `dsfont`, `bbm` and `mathtools`.

3.5 Page Geometry

Another important aspect of `phfnote` is the handling of page margins. Often the default page margins of the article class are quite narrow. While it is a good

typographical practice to avoid long lines, on occasion we prefer to have notes typeset with wider text. The general answer is the geometry package, which allows to set all margins in full detail.

The phfnote package provides some standard choices of options for the geometry package, which are adjusted according to the document font size, and whether the document is typeset in two columns.

If you want anything more complicated than what is provided by a default setting here, just use `pagegeomdefs=false` and invoke the geometry package directly with your preferred set of options.

```
pagegeomdefs=true
```

Include the geometry package, using the default settings or whatever is specified with the `pagegeom` option.

```
pagegeomdefs=false
```

Do not attempt to change the document margins, and don't load the geometry package.

```
nopagegeomdefs
```

Same as `pagegeomdefs=false`.

The page geometry predefined settings are the following.

```
pagegeom=default
```

Default settings. Not too wide, not too narrow. Settings vary according to single or double column setting, and according to default font point size.

```
pagegeom=narrow
```

Narrower style. For single-column documents, this is closer to the typographically-advertised-optimal of 50–80 characters per line, but it might look narrow to some.

```
pagegeom=wide
```

Wide, comfortable style. Wastes less paper.

```
pagegeom=xwide
```

Extra wide. Use if you pity trees.

```
pagegeom=bigmargin
```

Makes the margins asymmetric, so that a wide margin note can fit. This style is used in this package documentation, for example.

3.6 Section Headers Styling

The `phfnote` package provides some limited styling of section headers. The font, size and “compactness” of the headers can be adjusted with title options. But really, these options are quite basic. You should use `titlesec` or `sectsty` directly if you want anything serious.

The section headings are customized using the `sectsty` package. If this conflicts in your document, then use the `[secfmt={}]` package option to indicate that section headings should NOT be styled by this package. Then take care of section styling manually.

Package options may be used to customize the appearance of the section headings by specifying a list of attributes. When you combine arguments, make sure to put them in a braced group: `[secfmt={section,compact}]`. Beware that attributes are not merged between different occurrences of the `secfmt` keyword in the package options; the last occurrence defines all set attributes. If the `secfmt` package option is not given, then by default only the `section` attribute is set.

NOTE

Don't forget to include the attribute 'section' and/or 'paragraph' depending on which type of heading you want your settings to apply to. For example, `secfmt={sffamily}` has no effect, you need to use e.g. `secfmt={section,sffamily}`.

Available attributes are the following:

```
secfmt={section,...}
```

Use the `section` attribute to activate the styling of section-level headings, that is, `\section`, `\subsection` and `\subsubsection`.

```
secfmt={paragraph,...}
```

This attribute indicates that the styling should apply to paragraph-level headings as well (`\paragraph` and `\subparagraph`).

```
secfmt={compact,...}
```

Reduce the sizes of the section headings (if the section-level headings are styled, i.e. you need to specify the `section` attribute), giving the document a more “compact” appearance.

```
secfmt={larger,...}
```

Increase the sizes of the section headings. Suitable for longer documents or for small document font sizes.

`secfmt={secsquares,...}`

Display black squares on the left side of `\section`-level commands, making them stand out better. This is useful for documents (such as the present one) with several layers of sub-sections.

`secfmt={secnummargin,...}`

Display the section, subsection, and subsubsection numbering in the left margin and have the title occupy the full width of the text (such as for this document). If you want both `secsquares` and `secnummargin`, you must specify them in that order, or the black square may end up overlapping with the number.

`secfmt={rmfamily,...}`

Typeset headings in the regular roman font of the document, instead of trying to apply the **Palatino font**. This applies to section-level and/or paragraph-level headings, depending on which of the attributes `section` and/or `paragraph` have been specified.

`secfmt={sffamily,...}`

Typeset headings in a sans-serif font. The default document sans serif font is used. This applies to section-level and/or paragraph-level headings, depending on which of the attributes `section` and/or `paragraph` have been specified.

`secfmt={itpar,...}`

Typeset paragraph-level headings in italic.

`secfmt={blockpar,...}`

Change the paragraph-level headings not to be in “run-in” style, but to be typeset on their own line like section headings.

`secfmt={}`

Leave the argument empty to keep the original class styling; nothing will be overridden and the `sectsty` package is not loaded.

You can also directly modify the section heading style by redefining some macros. Note that these macros only affect those sectioning commands which we have decided to style, which is specified by the `section` and `paragraph` attributes to be specified in the `secfmt={...}` package option.

`\notesectionallfont` The macro `\notesectionallfont` is invoked for every sectioning command (for those which are styled, see the `section` and `paragraph` attributes). The macro `\notesectionallfont` internally invokes

`\notesectionallfontfamily` `\notesectionallfontfamily` to select which font family to use. The family should be given as the font code, e.g.: `pbk = Bookman`; `bch = Charter`; `ppl = Palatino`; `ptm = Adobe Times`; `phv = Adobe Helvetica`; `pcr = Adobe`

Courier; put = Utopia; cmr = Computer Modern Roman; cmss = CM Sans Serif; cmbr = CM Bright; google many more or look directly into the source of corresponding \LaTeX packages.

You may customize these either via attributes or by redefining them directly. Beware that if you redefine `\notesectionallfont` then you are responsible for honoring, or ignoring, the value of `\notesectionallfontfamily`.

<code>\notesectionfont</code>	These macros define the font commands to apply for the section heading corresponding to the given sectioning command. This macro is invoked after <code>\notesectionallfont</code> , which means that font definitions in these macros take precedence over those in <code>\notesectionallfont</code> .
<code>\notesubsectionfont</code>	
<code>\notesubsubsectionfont</code>	
<code>\noteparagraphfont</code>	
<code>\notesubparagraphfont</code>	
<code>\notesectionsetfonts</code>	The macro <code>\notesectionsetfonts</code> is a shorthand to set all section font definitions for the section-level commands <code>\section</code> , <code>\subsection</code> , and <code>\subsubsection</code> . For example,

```
\notesectionsetfonts{\Large}{\large}{\normalsize}
```

will set the font sizes for `\section`, `\subsection` and `\subsubsection` in this order.

<code>\noteparagraphsetfonts</code>	The macro <code>\noteparagraphsetfonts</code> is the corresponding shorthand for the paragraph-level commands. It takes two arguments, the font definitions to apply for headings of level <code>\paragraph</code> and <code>\subparagraph</code> .
-------------------------------------	---

3.7 Appearance of Paragraphs

Several presets may be set to define the appearance of paragraphs.

```
par=indent
```

Paragraphs are indented, bearing some similarity to the article class' default paragraph style.

```
par=skip
```

Paragraphs are separated by additional spacing, and not indented.

```
par=indentminiskip
```

Paragraphs are indented, but there is also a small space between each paragraph.

```
par=original
```

Do not modify the appearance of paragraphs, and leave the class default.

3.8 Adjusting Spacing of Lines and Words

The phfnote package also provides definitions to adjust spacing of lines and words.

This includes definitions to avoid overflowing words in the margin in case of long words.

```
spacingdefs=true
```

Apply adjustments to line and word spacing.

```
spacingdefs=false
```

Do not attempt any adjustments of line or word spacing.

```
nospacingdefs
```

Alias for spacingdefs=false.

3.9 Adjustments for Fonts

The phfnote package provides as well some adjustments for fonts to make some fonts look nicer.

Concretely, the Computer Modern Bright font is used as sans serif font instead of L^AT_EX's default sans serif font, and the more universal T1 font encoding is used instead of the default OT1.

```
fontdefs=true
```

Apply adjustments to fonts.

```
fontdefs=false
```

Do not apply adjustments to fonts.

```
nofontdefs
```

Same as fontdefs=false.

3.10 Footnote Figure Style

The footnotes' appearance can also be slightly enhanced.

```
footnotedefs=true
```

Changes the symbol appearance a little bit—the footnote number is smaller and typeset in boldface.


```
footnotedefs=false
```

Do not change the footnote appearance.

```
nofootnotedefs
```

Same as `footnotedefs=false`.

3.11 Hyperref Loading

There are many options for setting up the `hyperref` package, and often, the defaults (with boxed links) are pretty ugly in my opinion. Enable the `hyperrefdefs` feature of `phfnote` to alter the defaults to something I personally like better (dark blue links as in this document).

```
hyperrefdefs=true
```

Load the `hyperref` package, and set some sensible settings. Also ensures the `\email` and `\url` commands are made available.

```
hyperrefdefs=false
```

Do not load the `hyperref` package, do not set sensible settings.

```
nohyperrefdefs
```

Same as `hyperrefdefs=false`.

`\url` In order to typeset URLs, the `\url` command is made available from the package `url` (which is then linkified by `hyperref`). For example, you can type `\url{https://github.com/phfaist/}`.

`\email` A similar command allows to typeset e-mail addresses. The text is displayed as a hyperlink, which when clicked opens a e-mail composer to that address (via a `mailto:XXX` link). For example, try `\email{pulp_fiction@tarantino.com}`.

`\phfnotePdfLinkColor` The command `\phfnotePdfLinkColor` may be used to set the color of the links. It takes one argument, a color specification understood by the `xcolor` package. For example:

```
\phfnotePdfLinkColor{green!50!black}
```

NOTE

The package `xcolor` must be loaded for `\phfnotePdfLinkColor` to work. (The `xcolor` package is automatically loaded as part of a package set as long as you're not using the option `pkgset=none`; see subsection 3.4.)

3.12 Bibliography Definitions

This package also provides some definitions for the bibliography.

It sets the `naturemagdoi` style by default, which is a hacked (by yours truly) version of the `naturemag` style to include the journal name as a hyperlink (as in APS bibliography styles).

The bibliography is also typeset in a smaller font.

Finally, an entry in the table of contents is generated.

```
bibliographydefs=true
```

Load the `hyperref` package, and set some sensible settings. Also ensures the `\email` and `\url` commands are made available.

```
bibliographydefs=false
```

Do not load the `hyperref` package, do not set sensible settings.

```
nobibliographydefs
```

Same as `bibliographydefs=false`.

`\bibliography` The `\bibliographystyle` and `\bibliography` macros can be used as usual,
`\bibliographystyle` for example:

```
\bibliographystyle{apsrmp4-1} % optional  
\bibliography{mybibfile}
```

bearing in mind that if the `\bibliographystyle` command is not present, our custom `naturemagdoi` bibliography style is used.

3.13 URL Styles

As a bonus, the `phfnote` package provides an alternative set of URL styles to use with the `\url` and `\email` commands (see [subsection 3.11](#)).

All the styles described below typeset the URL in a slightly smaller size, so as to avoid a common issue with URLs that they tend to appear too large. Also, the tilde character is fixed so that it appears nicely, as in:

<https://people.phys.ethz.ch/~pfaist/>.

The URL style can be set with the command `\urlstyle{<name of style>}`.

`notett` typewriter font

`notesf` default sans serif font

<code>notesfss</code>	Computer Modern Sans Serif font
<code>noteitsf</code>	italic using default sans serif font
<code>noterm</code>	normal roman typeface
<code>noteit</code>	just italic typeface
<code>notesml</code>	just smaller than surrounding text

3.14 A `\notesmaller` Command

This general-purpose command is handy to typeset text smaller than its surrounding text, for when you don't know what size the surrounding text is typeset at. In some sense, this is a very very lightweight analogue of what the `reysize` package does. (This is used, for example, in our implementation of URL styles introduced in [subsection 3.13](#).)

<code>\notesmaller</code>	Set the font size to a fraction of the surrounding font size. The fraction may be specified as an optional argument. A fraction of 0.8 makes the text size 0.8 times that of the surrounding text, that is, smaller than the surrounding text. A value of 1 does not change the font size. If the fraction is not specified, the value stored in <code>\notesmallerfrac</code> is used.
<code>\notesmallerfrac</code>	The fraction by which <code>\notesmaller</code> typesets smaller text when no optional argument is given. You may redefine this command to set the default “smaller” size fraction.

3.15 Tools Mostly for Hackers

The `phfnote` package also provides some small hacks. They are documented further in [subsection 5.12](#). These are: a macro `\phfnoteHackSectionStarWithTOC` to hack into a command which generates a `\section*`, in order for that command to also generate a corresponding entry in the table of contents; and a pair of commands to save and restore \TeX definitions.

`\phfnoteSaveDefs`
`\phfnoteRestoreDefs`

■ 4 Summary of Package Options

`preset=<preset name>`

Load a preset specifying a predefined set of options for the general appearance of the document. See documentation in [subsection 2.2](#)

`title=<title style>`

Set the title style. Documentation in [subsection 3.1.1](#)

`abstract=<abstract attributes>`

Set the abstract style by specifying a comma-separated list of attributes. Don't forget to put the list of attributes within braces, [abstract={wide,noname,it}]. Documentation in [subsection 3.2](#)

`pkgset=<package set>`

Specify a standard set of L^AT_EX packages to load. See [subsection 3.4](#).

`pagegeomdefs=<true or false>`

Whether to care about page margins. `nopagegeomdefs` is synonym for `pagegeomdefs=false`.

`pagegeom=<geom style>`

Set a page margin style. Only has effect if `pagegeomdefs=true`. Options are documented in [subsection 3.5](#).

`secfmt=<section formatting attributes>`

A list of attributes defining how section (and possibly paragraph) headings should look like. See [subsection 3.6](#).

`par=<par style>`

Define how paragraphs should be spaced. Refer to [subsection 3.7](#).

`spacingdefs=<true or false>`

Adjust spacing of lines and words ([subsection 3.8](#)).

`fontdefs=<true or false>`

Adjust some fonts ([subsection 3.9](#)).

`footnotedefs=<true or false>`

Adjust slightly the appearance of footnotes. See [subsection 3.10](#).

`hyperrefdefs=<true or false>`

Load the `hyperref` package, and set some defaults settings. See [subsection 3.11](#).

`bibliographydefs=<true or false>`

Adjust the appearance and style of the bibliography. See [subsection 3.12](#).

TIP

To activate only a subset of features, use `preset=reset` and then enable only the features required. In this way, you can ensure that only those features which are explicitly specified are enabled.

■ 5 Implementation

Here comes the gory code.

Let's start by loading the kvoptions package, which we need to parse the package options. It's better to use xkeyval as backend, because the `\setkeys` by keyval is a little fragile: for example, it gets confused if, within a preset, we include a package or run a command which itself parses key-vals.

```
1 \RequirePackage{xkeyval}
2 \RequirePackage{kvoptions}
```

Also load etoolbox, for various utilities.

```
3 \RequirePackage{etoolbox}
```

5.1 Internal Generic Code

`\phfnote@internal@execattribs` An internal general-purpose macro to execute all definitions given in list of attributes.

Often, a list of attributes are given via a package option (e.g. for the abstract), and these attributes need to be executed, or implemented, in the order they are given. This macro takes care of that. Each possible attribute must be defined as a macro with a common prefix, to which the attribute is appended.

The arguments are:

- #1 = prefix to look for attributes (e.g. `noteabstract@attr@`);
- #2 = a human-readable name of what #1 represents, which is used in an error message in case the required attribute is not found (e.g. `{abstract attribute}`);
- #3 = the list of attributes specified by the user.

For example, `\phfnote@internal@execattribs{noteabstract@attr@}{abstract attribute}{noname,small}` causes the commands `\noteabstract@attr@noname` and `\noteabstract@attr@small` to be invoked, in this order.

```
4 \def\phfnote@internal@execattribs#1#2#3{%
5   \@for\next:=#3\do{%
6     \ifcsname #1\next\endcsname%
7     \csname #1\next\endcsname%
8     \else%
9     \PackageWarning{phfnote}{Unknown #2: '\next'. Ignoring.}
10    \fi
11   }
12 }
```

5.2 Title Styling

See [subsubsection 3.1.1](#) for a description of the styles and which features are available.

5.2.1 First, some common simple definitions for our different styles

<code>\notetitlefont</code>	These may be redefined to adapt the font of the title, author and date.
<code>\notetitleauthorfont</code>	
<code>\notetitledatefont</code>	13 <code>\newcommand{\notetitlefont}{\sffamily\bfseries}</code> 14 <code>\newcommand{\notetitleauthorfont}{}</code> 15 <code>\newcommand{\notetitledatefont}{\footnotesize}</code>
<code>\notetitlebelowspace</code>	These macros may be redefined to adjust spacing above and after the title. They are macros, not lengths, so they can be adjusted dynamically on the spot.
<code>\notetitletopspace</code>	16 <code>\newcommand{\notetitlebelowspace}{4mm}</code> 17 <code>\newcommand{\notetitletopspace}{-1.2cm}</code>
<code>\notetitlehrule</code>	Allow customization of the horizontal rule below the title. The macro <code>\notetitlehrule</code> expands to commands which generate the rule, such as “ <code>\hrule height 1pt</code> ”.
	18 <code>\newcommand{\notetitlehrule}{\hrule}</code>
<code>\notetitle@title</code>	Provide a “long” definition for <code>\title</code> , so that the title can have several paragraphs. Our style handles this by putting the title on several lines, and it can be useful depending on how you want to format the title. This macro will replace <code>\title</code> when a title style is actually selected in <code>\phfnote@do@notetitle</code> . 19 <code>\long\def\notetitle@title#1{\long\gdef\@title{#1}}</code>
<code>\phfnote@title@checksetspace</code>	Some of our title styles require the <code>setspace</code> package. This utility checks that this package is loaded, and generates an error otherwise. #1 = the current title style name; this is required only for the error message. 20 <code>\def\phfnote@title@checksetspace#1{%</code> 21 <code>\ifdefined\spacespace\else%</code> 22 <code>\PackageError{phfnote}{Note title style ‘#1’ requires the</code> 23 <code>‘setspace’ package to be loaded! Please load it, or use a</code> 24 <code>pkgset which loads it automatically}%</code> 25 <code>\fi%</code> 26 <code>}</code>

5.2.2 Implementation of `\thanks` and `\thanksmark`

Here we provide a few fixes for the implementation of `\thanks`, both for our main ‘default’ title style as well as for other simpler styles. Our implementation supports `\thanks[N]{...}` and `\thanksmark[N]` as for footnotes.

These newer implementations are only applied if one of our title styles is set. Otherwise, the class defaults are left (which may be needed, e.g., for REVTeX).

Implementation of `\thanks` and friends for our main ‘default’ title style

`\phfnote@setupthanksmppfootnote` Internal—called at the beginning of a minipage environment, it sets up necessary stuff to support `\thanks` notes within the minipage, in a single paragraph.

Some of this code was taken or really inspired directly from `latex.ltx`.

```
27 \def\phfnote@setupthanksmppfootnote{%
```

The `\thanks` macro is implemented as a `\footnote` in a minipage. So we hack into the ‘mpfootnote’ mechanism.

```
28 \def\thempfootnote{\arabic{mpfootnote}}%
29 \let\footnoterule\relax%
30 \let\thanks\footnote%
```

All footnote material is stored in a macro `\phfnote@mpfootmaterial`, initially empty:³

```
31 \def\phfnote@mpfootmaterial{}
```

and locally define `\@mpfootnotetext` to store the footnote content into that buffer,

```
32 \long\def\@mpfootnotetext##1{%
33   \protected@edef\@currentlabel%
34     {\csname p@mpfootnote\endcsname\@thefnmark}%
35   \protected@edef\@tmpa{\protect\phfnote@mymppfootnotemark{\@thefnmark}{##1}%
36     \protect\phfnote@mpfootnoteglue}%
37   \expandafter\g@addto@macro\expandafter\phfnote@mpfootmaterial%
38     \expandafter{\@tmpa}%
39   }%
```

Also provide `\thanksmark`, so that we can refer to other thanks/footnote-marks.

```
40 \def\thanksmark[##1]{\phfnote@mymppfootnotemark{##1}}%
41 }
```

³NOTE: this differs from how footnotes are usually treated (directly typeset into a vbox I think). Not sure what the side-effects might be. Because this is just for simple email/institute info/etc. in the title, hopefully this shouldn't have any serious consequences.

`\phfnote@finalizempfootnotes` Macro to call at the end of a minipage environment, to ensure that all `\footnote`'s (and thus `\thanks`'s) are properly formatted.

This simply takes all the tokens collected in `\phfnote@mpfootmaterial` (see just above), and typesets it in the `\@mpfootins` box. The latter is automatically typeset by the minipage in `\end{minipage}`.

```
42 \def\phfnote@finalizempfootnotes{%
43   \global\setbox\@mpfootins=\vbox{%
44     \parskip=0pt\parindent=0pt\parshape 1 0.04\textwidth 0.96\textwidth\relax%
45     \noindent\leavevmode%
46     \reset@font\footnotesize%
47     \phfnote@fmt@titlefootnotes%
48     \phfnote@mpfootmaterial}%
49 }
```

`\phfnote@fmt@titlefootnotes` Some formatting utilities which can be overridden if you know what you're doing.
`\phfnote@mypfootnotemark` `\phfnote@fmt@titlefootnotes` allows you to override the font in which the title-footnotes/thanks are typeset. `\phfnote@mypfootnotemark` is responsible for formatting its argument as a footnote mark, usually in superscript.
`\phfnote@mpfootnoteglue` `\phfnote@mpfootnoteglue` is the glue which is used between two footnote texts (as they are typeset in a single paragraph).

```
50 \def\phfnote@fmt@titlefootnotes{}
51 \def\phfnote@mypfootnotemark#1{\@textsuperscript{\normalfont#1}}
52 \def\phfnote@mpfootnoteglue{\hskip 1.2em plus 2em minus 0.5em\relax}
```

For those not using the main 'default' title style

We use \LaTeX 's own `\thanks` mechanism, however we patch on the possibility for using `\thanks [N] {text}` and `\thanksmark [N]` for overriding the number which is used.

`\notetitle@thanksmark` The `\thanksmark` is trivially implemented by `\footnotemark`. Very handy indeed.

Again, this macro is only made available as `\thanksmark` when a title style is set in `\phfnote@do@notetitle`.

```
53 \def\notetitle@thanksmark{\footnotemark}
```

Start by saving the old `\thanks` macro, just in case.

```
54 \let\phfnote@old@thanks\thanks
```

`\notetitle@thanks` Now, we need to extend \LaTeX 's `\thanks` to allow an optional argument as for footnotes. This macro will be renamed `\thanks` in `\phfnote@do@notetitle`.

Check whether there is an optional argument; if there is none we execute \LaTeX 's original thanks code (replicated here), otherwise, we specify the optional argument explicitly at the relevant location in \LaTeX 's implementation:

```
55 \def\notetitle@thanks{\@ifnextchar[\phfnote@thanks{\phfnote@thanks []}}%
56 \long\def\phfnote@thanks[#1]#2{%
57   \if\relax\detokenize{#1}\relax%
```

The optional argument is empty—just execute \LaTeX 's original `\thanks` code, replicated here:

```
58   \footnotemark%
59   \protected@xdef\@thanks{\@thanks\protect\footnotetext[\the\c@footnote]{#2}}%
```

Otherwise, execute \LaTeX 's original `\thanks` code, but with the optional argument inserted wherever needed:

```
60   \else% argument, pass on to sub-commands:
61     \footnotemark[#1]%
62     \protected@xdef\@thanks{\@thanks\protect\footnotetext[#1]{#2}}%
63   \fi%
64 }
```

5.2.3 Title Styles Definition

The title styles are documented in [subsection 3.1.1](#).

Title style: 'default'

Implementation our main 'default' title style. See [subsection 3.1.1](#).

`\notetitle@style@default` The default title style. Nothing mysterious, hopefully.

```
65 \newcommand{\notetitle@style@default}{%
66   \begingroup\par\raggedright%
67   \phfnote@setupthanksmppfootnote%
68   \vspace*{\notetitletopspace}%
69   \phfnote@title@checksetospace{default}%
70   \begin{minipage}{\textwidth}%
71     \begin{singlinspace}%
72       \parskip=0pt\parindent=0pt\relax%
73       {\let\phfnote@old@par\par%
74         \def\par{\phfnote@old@par%
75           \parskip=1.5ex\relax\parshape 1 0pt \textwidth\relax%
76           \noindent}%
77         \par%
78         \Large {\notetitlefont \@title}\par}%
79       \vskip 2mm\relax
```

```

80     \if\relax\detokenize\expandafter{\@author}\relax\else%
81     \par\parshape 1 0.04\textwidth 0.96\textwidth\relax%
82     {\notetitleauthorfont \@author}%
83     \vskip 2mm\relax%
84     \fi
85     \if\relax\detokenize\expandafter{\@date}\relax\else%
86     \par\parshape 1 0.04\textwidth 0.96\textwidth\relax%
87     {\notetitledatefont \@date}
88     \vskip 2mm\relax%
89     \fi
90     \global\let\@thanks\@empty%
91     \phfnote@finalizempfootnotes%
92     \end{singlespace}%
93     \end{minipage}\par%
94     \vspace*{2mm}%
95     \notetitlehrule\relax%
96     \par%
97     \endgroup%
98     \vskip\notetitlebelowspace\relax% don't change this, abstract needs to \removelastskip
99 }

```

Title style: 'small'

Implementation an alternate 'small' title style.

`\notetitle@style@small` The default title style. Nothing mysterious, hopefully.

```

100 \newcommand{\notetitle@style@small}{%
101   \begin{group}\par\raggedright%
102     \let\footnote\thanks%
103     \vspace*{\notetitletopspace}%
104     {\notetitlefont \@title}%
105     \hfill\makebox{\fontsize{9pt}{10pt}\selectfont {\notetitleauthorfont \@author}%
106       \hspace*{2mm}--\hspace*{2mm}{\emph{\notetitledatefont \@date}}}%
107     \vspace*{1mm}\notetitlehrule\relax\vspace*{1mm}%
108     \par%
109   \end{group}%
110   \vskip\notetitlebelowspace\relax% don't change this, abstract needs to \removelastskip
111 }

```

Title style: 'article'

Implementation the 'article' title style.

`\notetitle@style@article` The title style definition. Nothing mysterious, hopefully.

```

112 \newcommand{\notetitle@style@article}{%
113   \vspace*{-3em}%
114   \begin{group}

```

```

115   \centering
116   \let\footnote\thanks%
117   {\LARGE \@title \par}%
118   \vskip 1.5em%
119   {\large%
120     \lineskip .5em%
121     \begin{tabular}[t]{c}%
122       \@author%
123     \end{tabular}\par}%
124   \vskip 1.5em%
125   {\large \@date}%
126   \par%
127 \endgroup%
128 \par%
129 \vskip 2.5em\relax%
130 }

```

5.2.4 Plugging into \maketitle

Actually perform the definitions to make \maketitle produce the title with the given style. Specifically, we override \@maketitle. The latter is called internally by \maketitle, and the advantage of overriding \@maketitle only is that we inherit the mechanism provided by the style class to deal with two-column layouts.

`\phfnote@do@notetitle` This macro takes care of installing the correct title into the document, by overriding \@maketitle.

This macro is called later after processing the package options. Its argument #1 is the style name, e.g., default.

```

131 \def\phfnote@do@notetitle#1{

```

If we have an empty title style, then we leave default title provided by the class.

```

132 \if\relax\detokenize\expandafter{#1}\relax
133 \else

```

Otherwise, we have a title style to set. Do some checks that the given style is indeed defined.

```

134 \ifcsname notetitle@style@#1\endcsname
135 \def\phfnote@tmp@titsty{#1}%
136 \else
137 \PackageError{phfnote}{Unknown title style: '#1'.}{Unknown title
138 style: '#1'. Please consult the package documentation for available
139 styles.}
140 \def\phfnote@tmp@titsty{default}%
141 \fi

```

Apply new (default) definitions of `\thanks`, `\thanksmark` and `\title`. Do this here only, because this can clash with more complicated versions from, e.g., REVTeX.

```
142 \let\title\notetitle@title
143 \let\thanks\notetitle@thanks
144 \let\thanksmark\notetitle@thanksmark
```

Now, actually overload the title style by redefining `\@maketitle`.

```
145 \def\@maketitle{\csname notetitle@style@\phfnote@tmp@titsty\endcsname}
146 \fi
147 }
```

5.3 Abstract

Now we can take care of the abstract. Unlike the title styles, the abstract has a base implementation. Then, we may have attributes which change some parameters.

`notedefaultabstract` First, save the old environment `\begin{abstract}... \end{abstract}` provided by the class (if any).

```
148 \let\notedefaultabstract\abstract
149 \let\endnotedefaultabstract\endabstract
```

`\noteabstracttextfont` Macros which can be overridden to customize the abstract. See [subsection 3.2](#).

```
\noteabstractnamefont
\noteabstracttextwidth 150 \newcommand{\noteabstracttextfont}{}
\noteabstractafterspacing 151 \newcommand{\noteabstractnamefont}{\bfseries\small}
\noteabstractbeforepadding 152 \if@twocolumn
153 \newcommand\noteabstracttextwidth{\hspace}
154 \else
155 \newcommand{\noteabstracttextwidth}{0.9\hspace}
156 \fi
157 \newcommand\noteabstractafterspacing{1.5em}
158 \newcommand\noteabstractbeforepadding{1.5em}
```

`\noteabstract@nameline` Create the line which contains the title of the abstract, that is, the word “Abstract.” This can be overloaded, of course, for customization.

```
159 \def\noteabstract@nameline{
160   {\parskip=0pt\relax\par\centering\noteabstractnamefont%
161     \abstractname%
162     \par}\vskip 1ex\relax%
163 }
```

`noteabstract` The proper `noteabstract` environment.

```
164 \newenvironment{noteabstract}{%
165   \removeatstartskip%
166   \vspace{\noteabstractbeforespacing}%
167   \begin{group}%
168     \par\noindent\centering%
169     \begin{minipage}{\noteabstracttextwidth}%
170       \noteabstract@nameline%
171       \noteabstracttextfont%
172     }%
173   {%
174     \end{minipage}%
175     \par%
176   \endgroup%
177   \vspace{\noteabstractafterspacing}%
178 }
```

The abstract can be customized by the attributes. Here we define them:

```
179 \def\noteabstract@attr@wide{%
180   \def\noteabstracttextwidth{\textwidth}%
181 }
182 \def\noteabstract@attr@narrow{%
183   \if@twocolumn
184   \else
185     \def\noteabstracttextwidth{0.8\textwidth}%
186   \fi
187 }
188 \def\noteabstract@attr@noname{%
189   \def\noteabstract@nameline{}}\vspace*{1ex}}%
190 }
191 \def\noteabstract@attr@original{%
192   \let\abstract\notedefaultabstract
193   \let\endabstract\endnotedefaultabstract
194 }
195 \def\noteabstract@attr@small{%
196   \g@addto@macro\noteabstracttextfont{\small}%
197 }
198 \def\noteabstract@attr@compact{%
199   \renewcommand\noteabstractafterspacing{1ex}%
200   \renewcommand\noteabstractbeforespacing{1ex}%
201 }
202 \def\noteabstract@attr@it{%
203   \g@addto@macro\noteabstracttextfont{\itshape}%
204 }
```

`\phfnote@do@noteabstract` This helper both defines the abstract environment, and also sets the abstract attributes. This macro will be called according to the package options.

#1 = a comma-separated list of attributes.

```
205 \def\phfnote@do@noteabstract#1{
206   \let\abstract\noteabstract
207   \let\endabstract\endnoteabstract
208   \phfnote@internal@execattribs{noteabstract@attr@}{abstract attribute}{#1}
209 }
```

5.4 Page Geometry Settings

For the page geometry settings, we just have a bunch of styles which we define as macros. The macros just set up `\PassOptionsToPackage` for the geometry package. Then the correct macro will be selected according to the current `phfnote` package options.

The description of these settings are given in [subsection 3.5](#).

`\phfnote@pagegeomstyle@default` Default setting.

```
210 \def\phfnote@pagegeomstyle@default{
211   \if@twocolumn
212     \PassOptionsToPackage{hmargin=1in,vmargin=0.75in,includeheadfoot}{geometry}%
213   \else
214     % fix the margins a bit to make text wider
215     \ifcase\@ptsize% mods for 10 pt
216       \PassOptionsToPackage{hmargin=1.5in,vmargin=1.25in}{geometry}%
217     \or% mods for 11 pt
218       \PassOptionsToPackage{hmargin=1.5in,vmargin=1.25in}{geometry}%
219     \or% mods for 12 pt
220       \PassOptionsToPackage{hmargin=1.25in,vmargin=1.25in}{geometry}%
221     \fi%
222   \fi
223 }
```

`\phfnote@pagegeomstyle@narrow` Narrow style.

```
224 \def\phfnote@pagegeomstyle@narrow{
225   \if@twocolumn
226     \PassOptionsToPackage{hmargin=1.25in,vmargin=0.75in,includeheadfoot}{geometry}%
227   \else
228     % fix the margins a bit to make text wider
229     \ifcase\@ptsize% mods for 10 pt
230       \PassOptionsToPackage{hmargin=1.75in,vmargin=1.5in}{geometry}%
231     \or% mods for 11 pt
232       \PassOptionsToPackage{hmargin=1.75in,vmargin=1.5in}{geometry}%
233     \or% mods for 12 pt
234       \PassOptionsToPackage{hmargin=1.5in,vmargin=1.5in}{geometry}%
235     \fi%
236   \fi
```

```
237 }
```

`\phfnote@pagegeomstyle@wide` Wide style.

```
238 \def\phfnote@pagegeomstyle@wide{
239   \if@twocolumn
240     \PassOptionsToPackage{hmargin=0.75in,vmargin=0.75in,includeheadfoot}{geometry}%
241   \else
242     % fix the margins a bit to make text wider
243     \ifcase\@ptsize% mods for 10 pt
244       \PassOptionsToPackage{hmargin=1.25in,vmargin=1.25in}{geometry}%
245     \or% mods for 11 pt
246       \PassOptionsToPackage{hmargin=1.25in,vmargin=1.25in}{geometry}%
247     \or% mods for 12 pt
248       \PassOptionsToPackage{hmargin=1in,vmargin=1.25in}{geometry}%
249     \fi%
250   \fi
251 }
```

`\phfnote@pagegeomstyle@xwide` Extra wide.

```
252 \def\phfnote@pagegeomstyle@xwide{
253   \if@twocolumn
254     \PassOptionsToPackage{hmargin=0.5in,vmargin=0.5in,includeheadfoot}{geometry}%
255   \else
256     % fix the margins a bit to make text wider
257     \ifcase\@ptsize% mods for 10 pt
258       \PassOptionsToPackage{hmargin=1in,vmargin=1.25in}{geometry}%
259     \or% mods for 11 pt
260       \PassOptionsToPackage{hmargin=1in,vmargin=1.25in}{geometry}%
261     \or% mods for 12 pt
262       \PassOptionsToPackage{hmargin=0.75in,vmargin=1.25in}{geometry}%
263     \fi%
264   \fi
265 }
```

`\phfnote@pagegeomstyle@bigmargin` bigmargin style.

```
266 \def\phfnote@pagegeomstyle@bigmargin{%
267   \if@twocolumn
268     \PassOptionsToPackage{hmargin=1.5in,vmargin=0.75in,includeheadfoot}{geometry}%
269   \else
270     % fix the margins a bit to make text wider
271     \ifcase\@ptsize% mods for 10 pt
272       \PassOptionsToPackage{hmargin={2.25in,1.75in},vmargin=1.25in}{geometry}%
273     \or% mods for 11 pt
274       \PassOptionsToPackage{hmargin={2.25in,1.75in},vmargin=1.25in}{geometry}%
275     \or% mods for 12 pt
276       \PassOptionsToPackage{hmargin={2in,1.5in},vmargin=1.25in}{geometry}%
277     \fi%
```

```
278 \fi
279 }
```

`\phfnote@do@pagegeomdefs` Finally, provide a helper to set the page geometry. Just call the right macro.

```
280 \newcommand{\phfnote@do@pagegeomdefs}[1]{
281   \ifcsname phfnote@pagegeomstyle@#1\endcsname
282     \csname phfnote@pagegeomstyle@#1\endcsname
283   \else
284     \PackageWarning{phfnote}{Unknown page geometry style: '#1'!}
285   \fi
286
287   \RequirePackage{geometry}%
288 }
```

5.5 Text, Paragraph and Line Spacing

Text & Line Spacing

`\phfnote@do@spacing` Some cosmetic definitions to adjust line spacing. The line spacing is slightly adjusted according to font size to make the document more readable. Depending on whether the `setspace` package is loaded, we use it or go low-level with a redefinition of \LaTeX ' `\baselinestretch`. If the `captions` package is loaded, the figure captions' line spacing is also adjusted.

Also set an `\emergencystretch` so that lines get spaced out for underfull boxes, rather than overflowing far into the margin.

```
289 \def\phfnote@do@spacing{
290   \@ifpackageloaded{setspace}{
291     \def\phfnote@dostretch##1{%
292       \setstretch{##1}\phfnote@docaptionstretch{##1}}
293   }{
294     \def\phfnote@dostretch##1{%
295       \renewcommand\baselinestretch{##1}\phfnote@docaptionstretch{##1}}
296   }
297   \@ifpackageloaded{caption}{
298     \def\phfnote@docaptionstretch##1{\captionsetup{font={stretch=##1}}}
299   }{
300     \def\phfnote@docaptionstretch##1{\PackageWarning{phfnote}{Can't
301       set line spacing for captions, because the package 'caption'
302       is not loaded. Please load it before 'phfnote', or use an
303       appropriate (e.g. 'rich') pkgset which loads this package
304       automatically .}}
305   }
306   \if@twocolumn
307     \phfnote@dostretch{1.0} % leave default
308     \emergencystretch=3em\relax
```



```

309 \else
310   \ifcase \@ptsize% 10pt
311     \phfnote@dostretch{1.1}
312   \or% 11pt
313     \phfnote@dostretch{1.0} % 1.05? better 1.0...
314   \or% 12pt
315     \phfnote@dostretch{1.0} % 1.03? not really noticeable...
316   \fi
317   \emergencystretch=6em\relax
318 \fi
319 }

```

Paragraph Spacing Presets

Here again, we define several possibilities for paragraph settings as individual macros (see [subsection 3.7](#)). Depending on the package option, we execute the corresponding macro.

```

320 \def\phfnote@par@original{%
321 }
322 \def\phfnote@par@indent{%
323   \parindent=1.5em\relax
324   \parskip=0pt\relax
325 }
326 \def\phfnote@par@indentminiskip{%
327   \parindent=1.5em\relax
328   \parskip=0.3em plus 0.1em\relax
329 }
330 \def\phfnote@par@skip{%
331   \parindent=0pt\relax
332   \parskip=0.8em plus 0.2em minus 0.1em\relax
333 }

```

`\phfnote@do@par` Execute the given paragraph setting. The argument #1 is the setting, for example, `skip`.

```

334 \def\phfnote@do@par#1{%
335   \ifcsname phfnote@par@#1\endcsname
336     \csname phfnote@par@#1\endcsname
337   \else
338     \PackageWarning{phfnote}{Bad paragraph setting: #1. Leaving original}
339   \fi
340 }

```

5.6 Section Styling

Very limited support for styling section and paragraph headers ([subsection 3.6](#)). If you want anything serious, use `sectsty` or `titlesec` directly.

`\notesectionallfont` `\notesectionallfontfamily` Define the `\notesectionallfont` and `\notesectionallfontfamily`, which control the general font used in section headings.

```
341 \newcommand{\notesectionallfont}{%
342   \fontfamily{\notesectionallfontfamily}\fontseries{bx}\selectfont}
343 \newcommand{\notesectionallfontfamily}{ppl}
```

`\notesectionfont` `\notesubsectionfont` `\notesubsubsectionfont` `\noteparagraphfont` `\notesubparagraphfont` These macros are called for their respective sectioning command, after `\notesectionallfont` has been invoked. (Again, only for those sectioning commands which are styled by us.)

```
344 \newcommand{\notesectionfont}{\large}
345 \newcommand{\notesubsectionfont}{\normalsize}
346 \newcommand{\notesubsubsectionfont}{\small}
347 \newcommand{\noteparagraphfont}{\normalsize}
348 \newcommand{\notesubparagraphfont}{\normalsize}
```

`\notesectionsetfonts` `\noteparagraphsetfonts` Helpers to directly set the font commands for `\section`, `\subsection` and `\subsubsection` (with `\notesectionsetfonts`), and for `\paragraph` and `\subparagraph` (with `\noteparagraphsetfonts`).

```
349 \newcommand{\notesectionsetfonts}[3]{%
350   \renewcommand{\notesectionfont}{#1}%
351   \renewcommand{\notesubsectionfont}{#2}%
352   \renewcommand{\notesubsubsectionfont}{#3}%
353 }
354 \newcommand{\noteparagraphsetfonts}[2]{%
355   \renewcommand{\noteparagraphfont}{#1}%
356   \renewcommand{\notesubparagraphfont}{#2}%
357 }
```

Define the attributes which the user can set. See [subsection 3.6](#).

```
358 \def\phfnote@do@secfmt@section{
359   \RequirePackage{sectsty}
360   \sectionfont{\notesectionallfont\notesectionfont}
361   \subsectionfont{\notesectionallfont\notesubsectionfont}
362   \subsubsectionfont{\notesectionallfont\notesubsubsectionfont}
363 }
364 \def\phfnote@do@secfmt@paragraph{
365   \RequirePackage{sectsty}
366   \paragraphfont{\notesectionallfont\noteparagraphfont}
367   \subparagraphfont{\notesectionallfont\notesubparagraphfont}
368 }
369 \def\phfnote@do@secfmt@compact{
370   \notesectionsetfonts{\normalsize}{\small}{\small}
371 }
372 \def\phfnote@do@secfmt@larger{
373   \notesectionsetfonts{\Large}{\large}{\normalsize}
```

```

374 }
375
376 \def\phfnote@do@secfmt@secsquares{
377   \RequirePackage{amssymb}
378   \let\phfnote@secsquares@old@seccntformat\@seccntformat
379   \def\@seccntformat##1{%
380     \expandafter\ifx\csname ##1\endcsname\section\relax%
381     \unexpanded{\makebox[0pt][r]{\raisebox{0.15ex}{\{
382       \notesmaller[0.6]\ensuremath{\blacksquare}\}}}%
383       \hspace*{1.2ex}}}%
384     \fi%
385     \phfnote@secsquares@old@seccntformat{##1}}
386 }
387 \def\phfnote@do@secfmt@secnummargin{
388   \let\phfnote@secnummargin@old@seccntformat\@seccntformat
389   \def\@seccntformat##1{%
390     \protect\makebox[0pt][r]{\phfnote@secnummargin@old@seccntformat{##1}}}
391 }
392
393 \def\phfnote@do@secfmt@rmfamily{
394   \renewcommand\notesectionallfontfamily{\rmdefault}
395 }
396 \def\phfnote@do@secfmt@sffamily{
397   \renewcommand\notesectionallfontfamily{\sfdefault}
398 }
399 \def\phfnote@do@secfmt@itpar{
400   \def\noteparagraphfont{\normalfont\normalsize\itshape}
401   \def\notesubparagraphfont{\normalfont\normalsize\itshape}
402 }
403 \def\phfnote@do@secfmt@blockpar{
404   \let\phfnote@old@paragraph\paragraph
405   \def\paragraph##1{%
406     \phfnote@old@paragraph{##1}%
407     \hspace*{0pt}\par\nopagebreak% ugly hack!!
408   }
409 }

```

`\phfnote@do@secfmt` Actually perform the required styling, according to the package options given as argument. The argument is a comma-separated list of attributes specified by the user.

```

410 \def\phfnote@do@secfmt#1{%
411   \phfnote@internal@execattribs{phfnote@do@secfmt@}{section formatting preset}{#1}
412 }

```

5.7 L^AT_EX Package Sets

Define the package sets as macros. Depending on the user-specified options we load the corresponding one(s) (several may be specified).

See [subsection 3.4](#) for a description of what these package sets do.

<code>\phfnote@do@pkgset@none</code>	Macros which implement the package sets.	Each macro invokes
<code>\phfnote@do@pkgset@minimal</code>	<code>\RequirePackage</code> for the appropriate packages.	
<code>\phfnote@do@pkgset@rich</code>		
<code>\phfnote@do@pkgset@extended</code>		

```
413 \def\phfnote@do@pkgset@none{
414 }
415
416 \def\phfnote@do@pkgset@minimal{
417
418   \RequirePackage{amsmath}
419   \RequirePackage{amsfonts}
420   \RequirePackage{amssymb}
421   \RequirePackage{amsthm}
422
423   \RequirePackage{xcolor}
424
425 }
426
427 \def\phfnote@do@pkgset@rich{
428
429   \phfnote@do@pkgset@minimal
430
431   \RequirePackage{setspace}
432   \RequirePackage{caption}
433
434   \RequirePackage{microtype}
435
436   \PassOptionsToPackage{shortlabels}{enumitem}
437   \RequirePackage{enumitem}
438
439   \RequirePackage{graphicx}
440
441   \PassOptionsToPackage{T1}{fontenc}
442   \RequirePackage{fontenc}
443
444   \PassOptionsToPackage{utf8}{inputenc}
445   \RequirePackage{inputenc}
446 }
447
448 \def\phfnote@do@pkgset@extended{
449
450   \phfnote@do@pkgset@rich
451
452   \RequirePackage{float}
453
454   \RequirePackage{verbatim}
455
456   \PassOptionsToPackage{autostyle, autopunct=true}{csquotes}
457   \RequirePackage{csquotes}
```

```

458
459 \RequirePackage{dsfont}
460 \RequirePackage{bbm}
461 \RequirePackage{mathtools}
462
463 }

```

`\phfnote@do@pkgset` Finally, define the helper which will load the required package sets.

```

464 \def\phfnote@do@pkgset#1{
465 \phfnote@internal@execattribs{\phfnote@do@pkgset@}{package set}{#1}
466 }

```

5.8 Hyperref Support and Hyperlinks

NOTE

The name ‘docnotelinkcolor’ is historical and hard-coded in many other files I’ve used, so I’m DEFINITELY NOT changing it.

`\phfnote@do@pdfhyperrefdefs` Load the hyperref package and provide sensible defaults.

```

\email
\url 467 \newcommand{\phfnote@do@pdfhyperrefdefs}{%

```

Make sure a color-managing package is loaded, color or xcolor, and define our default color:

```

468 \phfnote@requirecolorpackage%
469 \definecolor{docnotelinkcolor}{rgb}{0,0,0.4}%

```

Load URL package, and save a version of `\url` which is not patched by hyperref:

```

470 \RequirePackage{url}%
471 \DeclareUrlCommand\phfnote@format@url{}%

```

Set up hyperref options:

```

472 \PassOptionsToPackage{bookmarks=true,backref=false}{hyperref}%
473 \RequirePackage{hyperref}%
474 %
475 \hypersetup{unicode=true,%
476   bookmarksnumbered=false,bookmarksopen=false,bookmarksopenlevel=1,%
477   breaklinks=true,pdfborder={0 0 0},colorlinks=true}%
478 \hypersetup{%
479   anchorcolor=docnotelinkcolor,citecolor=docnotelinkcolor,%
480   filecolor=docnotelinkcolor,linkcolor=docnotelinkcolor,%
481   menucolor=docnotelinkcolor,runcolor=docnotelinkcolor,%
482   urlcolor=docnotelinkcolor}%

```

Provide an `\email` command for specifying e-mails. Note that the `\url` command is already provided by the packages `url` and `hyperref`.

```
483 \let\email\phfnote@email%
```

And finally set a nicer default `\url/\email` style:

```
484 \urlstyle{notesf}%
485 }
```

`\phfnotePdfLinkColor` Set links color. Use as `\phfnotePdfLinkColor{<color>}`. Color may be any color name or specification recognized by the `xcolor` package.

```
486 \newcommand{\phfnotePdfLinkColor}[1]{%
487   \@ifpackageloaded{xcolor}{%
488     \colorlet{docnotelinkcolor}{#1}%
489   }{% else:
490     \PackageWarning{phfnote}{\protect\phfnotePdfLinkColor may only be
491       used if the package xcolor is loaded.}%
492   }%
493 }
```

`\phfnote@sanitize@url` Provide base macros to be able to build up `\email` command for emails and other URL-like commands which should sanitize their arguments.

`\phfnote@format@url`
`\phfnote@email`

Also prepare the command `\phfnote@email` which will be renamed `\email` in our `hyperref` package setup (see above).

```
494 \def\phfnote@sanitize@url{%
495   \catcode'\$12%
496   \catcode'\&12%
497   \catcode'\#12%
498   \catcode'\^12%
499   \catcode'\_12%
500   \catcode'\%12%
501   % \catcode'\^^J10%  newline = space
502   % \catcode'\^^M10%  newline = space
503   \relax%
504 }%
505 \providecommand\phfnote@format@url{\texttt}
506 \def\phfnote@email{\begingroup\phfnote@sanitize@url\phfnote@impl@email}%
507 \def\phfnote@impl@email@#1{\endgroup\href{mailto:#1}{\phfnote@format@url{#1}}}%

```

`\phfnote@requirecolorpackage` And finally define an internal utility to make sure that a color package (either color or xcolor) is loaded. If none are loaded, the xcolor package is loaded.

```
508 \def\phfnote@requirecolorpackage{%
509   \@ifpackageloaded{color}{%
510   }{%

```

```

511 \ifpackageloaded{xcolor}{%
512 }{%
513 \RequirePackage{xcolor}%
514 }%
515 }%
516 }

```

5.9 Cosmetic Font Definitions

`\phfnote@do@fontdefs` Minimalist cosmetic definition for fonts: load the T1 font encoding which is better. Also, use Computer Modern Bright as sans-serif font by default instead of Computer Modern Sans Serif.

```

517 \def\phfnote@do@fontdefs{
518
519 \PassOptionsToPackage{T1}{fontenc}
520 \RequirePackage{fontenc}
521
522 \renewcommand\sfddefault{cmbr}
523
524 }

```

5.10 Bibliography Stuff

Provide some fixes for the bibliography.

`\phfnote@bibstyle` Our default bibliography style is stored in `\phfnote@bibstyle`. By default, it's
`\phfnote@bibfont` our own hacked version of the naturemag style. The font in which to typeset the bibliography is stored in `\phfnote@bibfont`. By default, it's a little smaller than the main text.

```

525 \newcommand{\phfnote@bibstyle}{naturemagdoi}
526 \newcommand{\phfnote@bibfont}{\fontsize{9}{11}\selectfont}

```

`\phfnote@bibliography` These are a tentative implementation for `\bibliography`. The latter will be set to this implementation according to the user's package options.

```

527 \let\phfnote@old@bibliography\bibliography
528 \let\phfnote@old@bibliographystyle\bibliographystyle
529 \newcommand{\phfnote@bibliography}[1]{%
530 \begingroup%
531 \phfnote@bibfont%
532 \phfnote@old@bibliographystyle{\phfnote@bibstyle}%

```

Our hack: make sure that the next instance of `\section*` will generate a TOC entry. (See `\phfnoteHackSectionStarWithTOC`.)

```

533 \phfnoteHackSectionStarWithTOC%

```

Some special chars may appear in output of some ill-advised bibliography managers. Mostly the & symbol, such as in Taylor & Francis. We won't be needing a \LaTeX alignment operator here, so just make & a normal printable character (“other” catcode).

```
534 \catcode'\&=12\relax% normal char
```

Adjust the appearance of e-prints. We assume e-prints refer to the arXiv; here we generate a hyperlink and format them better.

```
535 \providecommand\epprint[2][\]{\href{http://arxiv.org/abs/##2}{arXiv:##2}}
```

Relay the call to the “old” `\bibliography` command to actually implement the bibliography.

```
536 \phfnote@old@bibliography{#1}%
537 \endgroup%
538 }
```

`\phfnote@bibliographystyle` Tentative implementation of `\bibliographystyle`. Just register the new style in an internal variable, so that the style is actually loaded in `\phfnote@bibliography`.

This will be renamed to replace `\bibliographystyle` later, according to package options.

```
539 \newcommand{\phfnote@bibliographystyle}[1]{%
540 \renewcommand{\phfnote@bibstyle}{#1}%
541 }
```

`\phfnote@do@bibliographydefs` Make our changes live. Will be called later according to package options.

```
542 \def\phfnote@do@bibliographydefs{%
543 \let\bibliographystyle\phfnote@bibliographystyle%
544 \let\bibliography\phfnote@bibliography%
545 }
```

5.11 Better Footnote Style

`\phfnote@do@footnotedefs` Adjust the formatting of footnotes so they look better. Again, this is called later according to the package options.

```
546 \def\phfnote@do@footnotedefs{
547 \let\phfnote@orig@makefnmark\@makefnmark
548 %% \def\@makefnmark{\hbox{\@textsuperscript{%
549 %% \normalfont\tiny\fontseries{sb}\selectfont\@thefnmark}}}
550 \def\@makefnmark{\hbox{\@textsuperscript{%
551 \normalfont\tiny\bfseries\@thefnmark}}}
```



```

552 %% \def\@makefnmark{\hbox{\@textsuperscript{%
553 %% \normalfont\scriptsize\bfseries\@thefnmark}}}% too large
554 }

```

5.12 Other Stand-Alone Definitions and Helpers

5.12.1 A `\notesmaller` command

`\notesmaller` Relative font size command. Makes the text a fraction smaller than its surroundings. The fraction is either given explicitly as optional argument (1.0=same size) or is by default set by `\notesmallerfrac`.

To impalement this, we exploit the fact that \TeX saves the current font size in the macro `\f@size`.

```

555 \newcommand\notesmaller[1][\notesmallerfrac]{%
556 \fontsize{#1\dimexpr\f@size pt\relax}{#1\dimexpr\f@baselineskip pt\relax}%
557 \selectfont\ignorespaces%
558 }

```

`\notesmallerfrac` Default fraction by which `\notesmaller` acts. Redefine to change defaults.

```

559 \def\notesmallerfrac{0.9}

```

5.12.2 Customized, “Inline,” Table of Contents

`\inlinetoc` Just a customized table of contents. Horizontal rules before and after, and spacing is adjusted, and no “Contents” title. The table of contents looks just like at the [top of this document](#). The command is described in [subsection 3.3](#).

We call `\starttoc` directly, bypassing the `\section*` included by `\tableofcontents` (see definition `\tableofcontents` in latex sources).

```

560 \newcommand{\inlinetoc}{%
561 \begingroup%
562 \vspace*{2mm}%
563 \hrule%
564 \vspace*{2mm}%
565 \parskip=1pt\relax%
566 \@starttoc{toc}%
567 \vspace*{4mm}%
568 \hrule%
569 \vspace*{6mm}%
570 \endgroup%
571 }

```

5.12.3 URL Styles

```
\url@notettstyle We also provide some URL styles. These can directly set with
\url@notesfstyle \urlstyle{<style-name>}.
\url@notesfssstyle
\url@noteitsfstyle 572 \def\url@notettstyle{%
\url@notermstyle 573 \def\UrlFont{\ttfamily\notesmaller}%
\url@noteitstyle 574 \phfnote@urlstyle@common%
\url@notesmlstyle 575 }
576 \def\url@notesfstyle{%
577 \def\UrlFont{\sffamily\notesmaller}%
578 \phfnote@urlstyle@common%
579 }
580 \def\url@notesfssstyle{%
581 \def\UrlFont{\fontfamily{cmss}\selectfont\notesmaller}%
582 \phfnote@urlstyle@common%
583 }
584 \def\url@noteitsfstyle{%
585 \def\UrlFont{\sffamily\itshape\notesmaller}%
586 \phfnote@urlstyle@common%
587 }
588 \def\url@notermstyle{%
589 \def\UrlFont{\rmfamily\notesmaller}%
590 \phfnote@urlstyle@common%
591 }
592 \def\url@noteitstyle{%
593 \def\UrlFont{\itshape\notesmaller}%
594 \phfnote@urlstyle@common%
595 }
596 \def\url@notesmlstyle{%
597 \def\UrlFont{\notesmaller}%
598 \phfnote@urlstyle@common%
599 }
```

`\phfnote@urlstyle@common` The following code is common to all our styles. We do an ugly hack in which the tilde character ('~') is fixed to the tilde char in the Adobe Times font (ptm code), so that it looks nicer and its alignment is correct.

```
600 \def\phfnote@url@tilde{\hbox{\fontfamily{ptm}\selectfont\textasciitilde}}
601 %%\def\phfnote@url@tilde{\raise-0.8ex\hbox{%
602 %% \kern-0.2ex\fontfamily{cmbr}\selectfont\textasciitilde}}
603 \def\phfnote@urlstyle@common{%
604 \def\UrlTildeSpecial{\do\~{\phfnote@url@tilde}}%
605 \let\Url@force@Tilde\UrlTildeSpecial%
606 }
```

5.12.4 Utility to Add TOC Entry For Starred Section

Here we provide an ugly hack which introduces an entry in the table of contents for `\section*` commands.

[Note: An existing way of adding the toc entry in these cases is to issue a `\addcontentsline` command before the relevant command (say `\bibliography`). However this is unreliable, because on page boundaries the `\addcontentsline` will pick up the previous page. This is why `\addcontentsline` should be issued right *after* the `\section*` command.]

WARNING

This command is truly a hack, don't apply it globally! It forces (locally) the `\section` command to be followed by a `*`! Do this within a group, just before a command which you are sure is invoking `\section*` (such as `\bibliography` in the article class).

`\phfnoteHackSectionStarWithTOC` Locally force `\section` to be followed by `*` and introduce an entry in the table of contents.

```
607 \def\phfnoteHackSectionStarWithTOC{%
608   \let\phfnote@old@section\section%
609   \def\section*##1{\phfnote@old@section*{##1}\addcontentsline{toc}{section}{##1}}%
610 }
```

`\phfnoteHackSectionStarWithTOCInCommand` Patches the given command (#1), which is known to invoke `\section*`, to locally first invoke `\phfnoteHackSectionStarWithTOC` and thus generate a TOC entry.

```
611 \def\phfnoteHackSectionStarWithTOCInCommand#1{%
612   \expandafter\let\csname phfnote@old@string#1\endcsname#1%
613   \gdef#1{%
614     \begingroup%
615     \phfnoteHackSectionStarWithTOC%
616     \csname phfnote@old@string#1\endcsname%
617     \endgroup%
618   }%
619 }
```

5.12.5 Hack to save & restore a set of commands

Exactly what it sounds like. You can store a set of commands, specified by their name, by specifying an identifier. The commands corresponding to a given identifier can then later be restored.

`\phfnoteSaveDefs` The command `\phfnoteSaveDefs{<identifier>}{<list of macro names>}` saves the current definitions of the given list of macro and associates them to the given identifier. The list of macros is specified as a comma-separated list of macro names.

```
620 \def\phfnoteSaveDefs#1#2{%
```

The macro `\phfnote@restoredefs@<identifier>` will store the code necessary to restore the macros.

```
621 \csgdef{phfnote@restoredefs@#1}{}%
```

Iterate over the macros we are supposed to store.

```
622 \def\@tmpa{#2}%
623 \@for\@next:=\@tmpa\do{%
```

For each macro we are supposed to store (whose name is given in `\next`), we `\let \phfnote@restoredefs@<identifier>@<macro-name>` store the current value of the macro.

```
624 \global\csletcs{phfnote@restoredefs@#1@\next}{\next}%
```

Then, we append to `\phfnote@restoredefs@<identifier>` the code necessary to restore this macro. That code is simply a `\cslet` instruction.

Recall that `\xappto` expands its second argument (as `\xdef` does), allowing us to expand the value of `\next`.

```
625 \expandafter\xappto\csname phfnote@restoredefs@#1\endcsname{%
626 \noexpand\csletcs{\next}{phfnote@restoredefs@#1@\next}%
627 }%
628 }%
629 }
```

`\phfnoteRestoreDefs` Restores the macro saved by `\phfnoteSaveDefs`. We simply execute the macro `\phfnote@restoredefs@<identifier>`, in which we duly stored the code necessary to restore all the saved macros.

```
630 \def\phfnoteRestoreDefs#1{%
631 \ifcsname phfnote@restoredefs@#1\endcsname%
632 \csname phfnote@restoredefs@#1\endcsname%
633 \else%
634 \PackageError{phfnote}{\string\phfnoteRestoreDefs: no such
635 definitions stored (#1)}
636 \fi%
637 }
```

5.12.6 A utility for verbatim stuff in arguments of other macros

FIXME: DOCUMENT ME!

A utility for using verbatim stuff in arguments of other macros—exploit `\detokenize`

```
638 \def\phfverb#1{%
639   \ifx\protect\relax%
640     \phfverbfmt{\detokenize{#1}\unskip}%
641   \else%
642     \noexpand\phfverb{\unexpanded{#1}}%
643   \fi%
644 }
645 \def\phfverbfmt#1{{\normalfont\texttt{#1}}}
```

5.13 Handle Package Options

5.13.1 Define and Parse Package Options

Initialization code for kvoptions for our package options. See [section 4](#).

```
646 \SetupKeyvalOptions{
647   family=phfnote,
648   prefix=phfnote@opt@
649 }
```

[title=...] The title style to use. See [subsubsection 3.1.1](#).

```
650 \DeclareStringOption[default]{title}
```

[abstract=...] Option for abstract attributes ([subsection 3.2](#)).

```
651 \DeclareStringOption[] {abstract}
```

[pkgset=...] Option for Package sets ([subsection 3.4](#))

```
652 \DeclareStringOption[rich]{pkgset}
```

[pagegeomdefs=...]
[pagegeom=...] Define the page geometry. See [subsection 3.5](#).

```
653 \DeclareBoolOption[true]{pagegeomdefs}
654 \DeclareComplementaryOption{nopagegeomdefs}{pagegeomdefs}
655 \DeclareStringOption[default]{pagegeom}
```

[secfmt=...] Styling of section headings. See [subsection 3.6](#).

```
656 \DeclareStringOption[section]{secfmt}
```

[par=...] How to treat paragraphs. See [subsection 3.7](#).

```
657 \DeclareStringOption[skip]{par}
```

[spacingdefs=...] Add definitions to adjust spacing of lines and words. See [subsection 3.8](#).

```
658 \DeclareBoolOption[true]{spacingdefs}
659 \DeclareComplementaryOption{nospacingdefs}{spacingdefs}
```

[fontdefs=...] Do some adjustments to the fonts. See [subsection 3.9](#).

```
660 \DeclareBoolOption[true]{fontdefs}
661 \DeclareComplementaryOption{nofontdefs}{fontdefs}
```

[footnotedefs=...] Adjustments for footnotes. See [subsection 3.10](#).

```
662 \DeclareBoolOption[true]{footnotedefs}
663 \DeclareComplementaryOption{nofootnotedefs}{footnotedefs}
```

[hyperrefdefs=...] Load hyperref and corresponding definitions. See [subsection 3.11](#).

```
664 \DeclareBoolOption[true]{hyperrefdefs}
665 \DeclareComplementaryOption{nohyperrefdefs}{hyperrefdefs}
```

[bibliographydefs=...] Adjustments for bibliography, including default style. See [subsection 3.12](#).

```
666 \DeclareBoolOption[true]{bibliographydefs}
667 \DeclareComplementaryOption{nobibliographydefs}{bibliographydefs}
```

[preset=...] Preset option. See [subsection 2.2](#).

```
668 \define@key{phfnote}{preset}{%
669   \ifcsname phfnote@preset@#1\endcsname%
670   \csname phfnote@preset@#1\endcsname%
671   \else%
672     \PackageError{phfnote}{Unknown preset: '#1'!}{You specified the
673       option 'preset=...' with an invalid value. Please look up the
674       package documentation corresponding to your version of phfnote
675       for possible values.}
676   \fi%
677 }
```

Provide the standard error message for unknown options.

```
678 \DeclareDefaultOption{%
679   \@unknownoptionerror
680 }
```

5.13.2 Define Global Presets

Define the global presets here. See [subsection 2.2](#) for a description of what these presets do.

`\phfnote@hook@atendload` A hook for presets to do stuff at the end of package load.

```
681 \def\phfnote@hook@atendload{}
```

`\phfnote@preset@article` Article preset.

```
682 \def\phfnote@preset@article{
683   \def\phfnote@opt@title{article}
684   \def\phfnote@opt@par{indent}
685   \def\phfnote@opt@pagegeom{default}
686 }
```

`\phfnote@presetcommon@xnote` Specify some common definitions for all our *note preset styles. The optional argument is the URL style to set.

```
687 \newcommand\phfnote@presetcommon@xnote[1][noteitsf]{
688   \def\phfnote@opt@title{default}
689   \def\phfnote@opt@par{skip}
690   \phfnote@opt@pagegeomdefstrue
691   \def\phfnote@opt@pagegeom{wide}
692   \setlength{\footnotesep}{5pt}
693   \g@addto@macro\phfnote@hook@atendload{
694     \ifdefined\urlstyle
695       \urlstyle{#1}
696     \fi
697   }
698 }
```

`\phfnote@preset@sfnote` Define the different *note styles.

```
\phfnote@preset@sfssnote
\phfnote@preset@opensansnote
\phfnote@preset@utopianote
\phfnote@preset@mmynote
699 \def\phfnote@preset@sfnote{
700   \phfnote@presetcommon@xnote
701   \phfnote@opt@footnotedefstrue
702   \phfnote@opt@fontdefstrue
703   \renewcommand\familydefault{\sfdefault}
704   \renewcommand{\notesectionallfontfamily}{\sfdefault}
705 }
706 \def\phfnote@preset@sfssnote{
```

set up all the settings as for sfnote ...

```
707 \phfnote@preset@sfnote
```

...but override:

```
708 \phfnote@opt@fontdefsfalse
709 \PassOptionsToPackage{T1}{fontenc}
710 \RequirePackage{fontenc}
711 \renewcommand\sfddefault{cmss}
712 }
713 \def\phfnote@preset@opensansnote{
```

set up all the settings as for sfnote ...

```
714 \phfnote@preset@sfnote
```

...but override:

```
715 \phfnote@opt@fontdefsfalse
716 \PassOptionsToPackage{T1}{fontenc}
717 \RequirePackage{fontenc}
718 \PassOptionsToPackage{default,osfigures,scale=0.9}{opensans}
719 \RequirePackage{opensans}
720 }
721 \def\phfnote@preset@utopianote{
722 \phfnote@presetcommon@xnote[noteit]
723 \phfnote@opt@fontdefsfalse
724 \PassOptionsToPackage{T1}{fontenc}
725 \RequirePackage{fontenc}
726 \RequirePackage{fourier}
727 \renewcommand{\notesectionallfontfamily}{put}
728 \renewcommand{\notetitlefont}{\bfseries}
729 \renewcommand\sfddefault{phv}
730 }
731 \def\phfnote@preset@mmynote{
732 \phfnote@presetcommon@xnote[noteit]
733 \phfnote@opt@footnotedefsfalse
734 \phfnote@opt@fontdefsfalse
735 \PassOptionsToPackage{T1}{fontenc}
736 \RequirePackage{fontenc}
737 \renewcommand{\notesectionallfontfamily}{\sfddefault}
```

Require these packages AFTER the default package set, because some symbols may be defined in package sets, and I've had problems with re-definitions etc... anyway this seems to work this way:

```
738 \g@addto@macro\phfnote@hook@atendload{
739 \RequirePackage{MnSymbol}
740 \PassOptionsToPackage{medfamily,textosf,mathlf,minionint,footnotefigures}{MinionPro}
741 \RequirePackage{MinionPro}
742 \PassOptionsToPackage{medfamily}{MyriadPro}
743 \RequirePackage{MyriadPro}
744 }
745 }
```


`\phfnote@preset@pkgdoc` Preset for a package documentation.

Start by setting the same settings as for other Xnote presets.

```
746 \def\phfnote@preset@pkgdoc{
747   \phfnote@presetcommon@xnote[noteit]
748   \phfnote@opt@fontdefsfalse
```

Then set up the font, which is done in a separate macro `\phfnote@pkgdoc@setupfont` in case individual documents would like more specific settings. (For example, some packages may want a different math font.)

```
749 \phfnote@pkgdoc@setupfont
```

Finally, set up general appearance.

```
750 \def\phfnote@opt@secfmt{section,paragraph,itpar,blockpar,larger,secsquares,secnummargin}
751 \def\phfnote@opt@pagegeom{bigmargin}
752 \def\phfnote@opt@abstract{noname}
753 }
```

Also provide a helper macro which is to load the font packages we want. By default, we use Utopia fonts via the `fourier` package, but some package documentations may want a different math font. Override `\phfnote@pkgdoc@setupfont` to adjust the whole font set-up, or `\phfnote@pkgdoc@setupmainfont` to adjust only the main document font.

```
754 \providecommand\phfnote@pkgdoc@setupfont{
755   \PassOptionsToPackage{T1}{fontenc}
756   \RequirePackage{fontenc}
757   \phfnote@pkgdoc@setupmainfont
758   \renewcommand{\notesectionallfontfamily}{put}
759   \renewcommand{\notetitlefont}{\bfseries}
760   \IfFileExists{opensans.sty}{\PackageError{phfnote}{Font OpenSans is not
761     available (need 'opensans' package)}{Please install the opensans
762     package, which provides the OpenSans font.}}
763   \def\opensans@scale{s*[0.85]}
764   \renewcommand{\sfdefault}{fosj}
765 }
766 \providecommand\phfnote@pkgdoc@setupmainfont{\RequirePackage{fourier}}
```

`\phfnote@preset@xpkgdoc` Same as `preset=pkgdoc`, but also provide some handy hacks and commands.

```
767 \def\phfnote@preset@xpkgdoc{
768   \phfnote@preset@pkgdoc
```

Include the `verbdef` package, because it's always useful.

```
769 \RequirePackage{verbdef}
```

Some patching first: Patch up `\PrintChanges` and `\PrintIndex`, if they are defined (for if we are using the `ltxdoc` package for latex package documentation). We want these to generate an entry in the table of contents. Also provide the utility `\PrintChangesAndIndex`, which calls both `\PrintChanges` and `\PrintIndex` with some additional spacing.

```

770 \ifdefined\PrintChanges
771   \phfnoteHackSectionStarWithTOCInCommand\PrintChanges
772 \fi
773 \ifdefined\PrintIndex
774   \phfnoteHackSectionStarWithTOCInCommand\PrintIndex
775 \fi
776 \def\PrintChangesAndIndexSpacing{\vspace{3cm plus 2cm minus 2cm}}
777 \def\PrintChangesAndIndex{\PrintChangesAndIndexSpacing\PrintChanges
778   \PrintChangesAndIndexSpacing\PrintIndex}

```

Set the index to TWO columns only (three is too tight).

```

779 \ifdefined\c@IndexColumns
780   \setcounter{IndexColumns}{2}
781 \fi

```

And set the glossary, that is, the list of changes history to single-column. For this, renew the environment completely to remove the `multicols` environment.

```

782 \let\phfnote@xpkgdoc@old@theglossary\theglossary
783 \let\phfnote@xpkgdoc@old@endtheglossary@endtheglossary
784 \renewenvironment{theglossary}{%
785   \glossary@prologue%
786   \GlossaryParms \let\item\@idxitem \ignorespaces}
787 {}

```

Hyperref: No “default” hyperref definitions, we’ll use `hyperdoc` instead.

```

788 \phfnote@opt@hyperrefdefsfalse
789 \g@addto@macro\phfnote@hook@atendload{
790   \definecolor{docnotelinkcolor}{rgb}{0,0,0.4}%
791   \RequirePackage{url}%
792   \DeclareUrlCommand\phfnote@format@url{%
793     \RequirePackage{hypdoc}
794     %
795     \hypersetup{bookmarks=true,backref=false,unicode=true,%
796       bookmarksnumbered=false,bookmarksopen=false,bookmarksopenlevel=1,%
797       breaklinks=true,pdfborder={0 0 0},colorlinks=true}%
798   \hypersetup{%
799     anchorcolor=docnotelinkcolor,citecolor=docnotelinkcolor,%
800     filecolor=docnotelinkcolor,linkcolor=docnotelinkcolor,%
801     menucolor=docnotelinkcolor,runcolor=docnotelinkcolor,%
802     urlcolor=docnotelinkcolor}%
803   \let\email\phfnote@email%
804   \urlstyle{noteit}

```

```
805 }
```

Provide Macro: `\pkgname{<package name>}` to format a package name. Also place it in the general index. This command is robust and can be used in section titles etc.

```
806 \def\pkgname##1{%
807   \pkgnamefmt{##1}%
808   \index{##1=\pkgnamefmt{##1}|hyperpage}%
809   \index{packages:>##1=\pkgnamefmt{##1}|hyperpage}%
810 }
811 \robustify\pkgname
812 \def\pkgnamefmt##1{\textsf{##1}}
813 \robustify\pkgnamefmt
```

Provide Macros: `\changed` and `\changedref text`, with more advanced support for displaying changes in package functionality or API.

First, we need a counter for the x-ref system.

```
814 \newcounter{phfnotechanged}
```

Mark changes in the implementation section of the package documentation with the command `\changed[<label name>]{<v1.0>}{<2016/05/22>}{<description>}`. This command automatically adds the change to the package's change history list, and allows you to refer to this change anywhere else in the package doc with `\changedref text`.

```
815 \newcommand*\changed[4] [] {%
```

First, if no label is given as optional argument, then just display the change and add it to the package changes list.

```
816   \if\relax\detokenize{##1}\relax%
817     \changedtextfmt{##2}{##3}{##4}%
818     \changes{##2}{##3}{##4}%
819   \else%
```

If a label name is provided as optional argument, then we need to write some stuff to the `.aux` file to make the change visible in the whole document.

```
820     \protected@edef\phfnotechanged@tmpa{##2}{##3}{##4}%
821     \immediate\write\@auxout{\string\phfnote@changed@set%
822       {##1}{\expandonce\phfnotechanged@tmpa}}%
823     \par\hspace*{0pt}\refstepcounter{phfnotechanged}\label{phfnotechanged:##1}%
824     \begingroup\let\phfnote@changedref text@par\relax
825       \changedref text{##1}%
826     \endgroup
827     \changes{##2}{##3}{\hyperref [phfnotechanged:##1]{##4}}%
828   \fi
829 }
```

```

830 \def\phfnote@changed@set##1{%
831   \expandafter\gdef\csname phfnote@changed@lbl@##1\endcsname%
832 }

```

When you document changes with the help of `\changed`, you may refer to any specific change from anywhere else in the package doc with the help of `\changedreftext{<label name>}`.

```

833 \def\phfnote@changedreftext@par{\par}
834 \newcommand*\changedreftext [1] {%
835   \phfnote@changedreftext@par%
836   \ifcsname phfnote@changed@lbl@##1\endcsname
837     \hyperref [phfnotechanged:##1] {%
838       \expandafter\expandafter\expandafter\changedtextfmt%
839       \csname phfnote@changed@lbl@##1\endcsname
840     }
841   \else
842     \hyperref [phfnotechanged:##1] {%
843       \changedtextfmt{???}{???}{[\textbf{missing ref}]}%
844     }%
845   \fi
846   \par
847 }

```

The macro `\changedtextfmt{<v1.0>}{<2016/05/22>}{<description>}` takes care of formatting the change on the spot.

```

848 \newcommand*\changedtextfmt [3] {%
849   \textit{Changed in {##1}\kern 0.3ex\relax{##2}}:} ##3.
850 }

```

Provide environment `pkgoptions`: Set up an elaborate environment (based on a description environment) to describe package options.

```

851 \RequirePackage{enumitem}
852 \newlist{pkgoptions}{description}{1}
853 \setlist [pkgoptions] {font=\pkgoptionfmt [{\vspace*{5pt}}], style=nextline}

```

But patch the `pkgoptions`' `\item` command, so that it puts an additional pair of braces around its argument. In this way, the `font=` attribute for the list sees the full label as its next token, and can be used as a macro argument. (This is not needed for newer versions of `enumitem`.)

```

854 \apptocmd\pkgoptions{\let\pkgoptions@old@item\item%
855   \def\item{\@ifnextchar[\pkgoptions@item@\pkgoptions@item@@}%
856   \def\pkgoptions@item@[##1]{\pkgoptions@old@item[{{##1}}]}%
857   \def\pkgoptions@item@@{\PackageWarning{phfnote}{\pkgoptions}: you must
858     specify label to \string\item as \string\item[label].}%
859   \pkgoptions@old@item}%
860 }{\PackageWarning{phfnote}{preset xpkgdoc: Failed to patch command
861   \string\pkgoptions}}

```

```
862 \def\pkgoptionscombineitem{\leavevmode\vspace{\dimexpr-\baselineskip-\parskip-\itemsep\relax}
```

For convenience, also provide a `\meta`-like command for boolean arguments (true or false). ‘`\metatruefalsearg`’ typesets as ‘`<true | false>`’.

```
863 \def\metatruefalsearg{\meta{\phfverb{true} $\mid$ \phfverb{false}}}
```

Include also a command to format a package option. Puts the option in a box in typewriter text style, and indexes it. The optional argument is meant to be internal—it adds commands after the displayed text (use it to add, e.g. spacing).

When indexing the packages, make sure to remove the protective braces if any.

```
864 \newcommand\pkgoptionfmt[2] []{%-
865   \begingroup\let\meta\pkgoptfmt@meta\fbbox{\normalfont\ttfamily ##2}\endgroup%
866   \expandafter\phfnote@pkgdoc@index\expandafter{\@firstofone ##2}%
867   ##1}
868 \let\pkgopt@save@meta\meta
869 \def\pkgoptfmt@meta##1{\begingroup\normalfont\itshape\pkgopt@save@meta{##1}\endgroup}
```

Whenever a package option is formatted with `\pkgoptionfmt`, it is placed in the index. Because package options may be of the form `key=val`, we want to split keys from values and put them independently in the index. This is done by entering a \TeX group, and using an `\lccode` trick: the code is prepared to iterate over a list of comma-separated stuff, but then the “lowercase” version of that code is executed instead, where the `=`’s have been replaced by `,`’s.

```
870 \def\phfnote@pkgdoc@index##1{%
871   \begingroup\lccode'\ = '\ ,\relax%
872   \def\x{\lowercase{\def\@tmpa{##1}}}%
873   \x%
874   \let\meta@gobble%
875   \let\marg@gobble%
876   \let\oarg@gobble%
877   \let\parg@gobble%
878   \let\pkgoptattrib\@firstofone%
879   \let\pkgoptattribnodots\@firstofone%
880   \let\pkgoptattribempty\@empty%
881   \def\handleitemindex####1{%
882     \edef\@tmpc{####1}%
883     \if\relax\detokenize\expandafter{\@tmpc}\relax\else%
884       \edef\@tmpb{\expandonce\@tmpc=\string\verb!*\+\expandonce\@tmpc+ (\pkgoptname)|hyp
885       \expandafter\index\@tmpb%
886       \edef\@tmpb{\packageoptionsname:>\expandonce\@tmpc=\string\verb!*\+\expandonce\@tmp
887       \expandafter\index\@tmpb%
888       \fi%
889     }%
890     \def\@tmpc{\forcsvlist{\handleitemindex}}%
891     \expandafter\@tmpc\expandafter{\@tmpa}%
892   \endgroup%
893 }
```

```

894 \def\pkgoptname{pkg. opt.}
895 \def\packageoptionsname{package options}

```

Provide environment cmdoptions: hijack the pkgoptions environment to do the same thing, except we place the items in the index under “command options” instead of “package options.”

```

896 \def\cmdoptions{\begingroup\setcmdnotpkgoptions
897   \pkgoptions}
898 \def\endcmdoptions{\endpkgoptions\endgroup}
899 \newcommand\cmdoptionfmt[2][\begingroup\setcmdnotpkgoptions
900   \pkgoptionfmt[{\##1}]{\##2}\endgroup}
901 \def\cmdoptname{cmd. opt.}
902 \def\commandoptionsname{command options}
903 \def\setcmdnotpkgoptions{\let\pkgoptname\cmdoptname
904   \let\packageoptionsname\commandoptionsname
905   \let\fbbox\cmdoptionsfbx}
906 \def\cmdoptionsfbx##1{\ensuremath{\underline{\{\text{\##1}}}}}

```

Provide the `\pkgoptattrib` command, which typesets its argument as `\{arg, ... \}`—useful to typeset attributes such as in [subsection 3.2](#). The variant `\pkgoptattribnodots{arg}` typesets `\{arg\}` while `\pkgoptattribempty` expands to `{}`.

```

907 \def\pkgoptattrib##1{\{\##1, ... \}}
908 \def\pkgoptattribnodots##1{\{\##1\}}
909 \def\pkgoptattribempty{\{\}}

```

Colorful boxes: environments pkgnote, pkgwarning, and pkgtip. Now, load the `tcolorbox` package to provide visual “Note,” “Warning,” and “Tip” boxes. Because `tcolorbox` includes the `verbatim` package which messes up the `verbatim` environment in latex dtx files (for which source lines all start with a % which needs to be stripped), we save the `verbatim`-related commands, and restore them after the interfering packages have been loaded.

```

910 \phfnoteSaveDefs{verbatimstuff}{%
911   verbatim,@verbatim,@xverbatim,@sxverbatim,endverbatim}
912 \usepackage{tcolorbox}
913 \newtcolorbox{pkgnote}{
914   colback=blue!5!white,
915   colframe=blue!5!white,
916   coltitle=blue!50!black,
917   toptitle=1.5ex,
918   fonttitle=\bfseries,
919   title={NOTE}
920 }
921 \newtcolorbox{pkgwarning}{
922   colback=red!5!white,
923   colframe=red!5!white,
924   coltitle=red!50!black,

```

```

925     toptitle=1.5ex,
926     fonttitle=\bfseries,
927     title={WARNING}
928 }
929 \newtcolorbox{pkgtip}{
930     colback=green!5!white,
931     colframe=green!5!white,
932     coltitle=green!50!black,
933     toptitle=1.5ex,
934     fonttitle=\bfseries,
935     title={TIP}
936 }
937 \phfnoteRestoreDefs{verbatimstuff}

```

Common title stuff:

```

938 \def\phfqitltxPkgTitle##1{The \pkgname{##1} package\thanks{\itshape
939     This document corresponds to \pkgname{##1}~\fileversion, dated \filedate. It
940     is part of the
941     \href{https://github.com/phfaist/phfqitltx/}{\pkgname{phfqitltx}} package
942     suite, see \url{https://github.com/phfaist/phfqitltx}.}}

```

Utility to parse package file date into “\today”-style date: invoke as
`\date{\pkgfmtdate\filedate}`.

```

943 \def\pkgfmtdate##1{%
944     \edef\pkgfmtdate@thedata{##1}%
945     \expandafter\pkgfmtdate@next\pkgfmtdate@thedata\@nil%
946 }
947 \def\pkgfmtdate@next##1/##2/##3\@nil{% YYYY/MM/DD
948     \ifcase ##2 \or January\or February\or March\or April\or May%
949     \or June\or July\or August\or September\or October\or November\or December\fi\space ##3,%
950     \space ##1}
951 \robustify\pkgfmtdate@next
952 }

```

`\phfnote@preset@reset` Finally, the reset preset:

```

953 \def\phfnote@preset@reset{
954     \def\phfnote@opt@pkgset{none}
955     \def\phfnote@opt@title{}
956     \phfnote@opt@pagegeomdefsfalse
957     \phfnote@opt@spacingdefsfalse
958     \def\phfnote@opt@par{original}
959     \def\phfnote@opt@abstract{original}
960     \phfnote@opt@hyperrefdefsfalse
961     \phfnote@opt@fontdefsfalse
962     \def\phfnote@opt@secfmt{}
963     \phfnote@opt@bibliographydefsfalse
964     \phfnote@opt@footnotedefsfalse

```

WARNING

SELF-NOTE: DO NOT FORGET TO ADD HERE RESET COMMANDS FOR ANY NEW OPTION THAT WE PROVIDE IN THE FUTURE.

```
965 }
```

5.13.3 Finally, Process and Execute the Package Options

Process the options:

```
966 \ProcessKeyvalOptions*
```

Take action according to the user options.

```
967 \phfnote@do@pkgset{\phfnote@opt@pkgset}
968
969 \phfnote@do@notetitle{\phfnote@opt@title}
970
971 \phfnote@do@noteabstract{\phfnote@opt@abstract}
972
973 \phfnote@do@secfmt{\phfnote@opt@secfmt}
974
975 \ifphfnote@opt@pagegeomdefs
976   \phfnote@do@pagegeomdefs{\phfnote@opt@pagegeom}
977 \fi
978
979 \ifphfnote@opt@spacingdefs
980   \phfnote@do@spacing
981 \fi
982
983 \phfnote@do@par{\phfnote@opt@par}
984
985 \ifphfnote@opt@hyperrefdefs
986   \phfnote@do@pdfhyperrefdefs
987 \fi
988
989 \ifphfnote@opt@fontdefs
990   \phfnote@do@fontdefs
991 \fi
992
993 \ifphfnote@opt@bibliographydefs
994   \phfnote@do@bibliographydefs
995 \fi
996
997 \ifphfnote@opt@footnotedefs
998   \phfnote@do@footnotedefs
999 \fi
```


Finally, execute the hook we set up for definitions at the end of the package loading:

```
1000 \phfnote@hook@atendload
```

Change History

v1.0

General: Initial version 1

Index

Numbers written in *italics* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in **roman** refer to the code lines where the entry is used.

Symbols	
\#	497
\\$	495
\%	500
\&	496, 534
\,	871
\=	871
\@author	80, 82, 105, 122
\@auxout	821
\@currentlabel	33
\@date	85, 87, 106, 125
\@empty	90, 880
\@firstofone	866, 878, 879
\@for	5, 623
\@gobble	874, 875, 876, 877
\@idxitem	786
\@ifnextchar	55, 855
\@ifpackageloaded	290, 297, 487, 509, 511
\@makefnmark	547, 548, 550, 552
\@maketitle	145
\@mpfootins	43
\@mpfootnotetext	32
\@nil	945, 947
\@optsize ..	215, 229, 243, 257, 271, 310
\@secntformat ..	378, 379, 388, 389
\@starttoc	566
\@textsuperscript ..	51, 548, 550, 552
\@thanks	59, 62, 90
\@thefnmark	34, 35, 549, 551, 553
\@title	19, 78, 104, 117
\@tmpa	35, 38, 622, 623, 872, 891

<code>\@tmpb</code>	884, 885, 886, 887	<code>\changedreftext</code>	825, 834
<code>\@tmpc</code>	882, 883, 884, 886, 890, 891	<code>\changedtextfmt</code> ...	817, 838, 843, 848
<code>\@unknownoptionerror</code>	679	<code>\changes</code>	818, 827
<code>\{</code>	907, 908, 909	<code>\cmdoptionfmt</code>	899
<code>\}</code>	907, 908, 909	<code>\cmdoptions</code>	896
<code>\^</code>	498, 501, 502	<code>\cmdoptionsfbox</code>	905, 906
<code>_</code>	499	<code>\cmdoptname</code>	901, 903
<code>\~</code>	604	<code>color</code>	38
A			
<code>\abstract</code>	148, 192, 206	<code>\colorlet</code>	488
<code>abstract (pkg. opt.)</code>	10, 20	<code>\commandoptionsname</code>	902, 904
<code>abstract (environment)</code>	9	<code>compact (pkg. opt.)</code>	10, 13
<code>\abstractname</code>	161	<code>\csgdef</code>	621
<code>\addcontentsline</code>	609	<code>\csletcs</code>	624, 626
<code>amsfonts</code>	11	<code>\csname</code> ... 7, 34, 145, 282, 336, 380,	
<code>amsmath</code>	11	612, 616, 625, 632, 670, 831, 839	
<code>amssymb</code>	11	<code>csquotes</code>	11
<code>amsthm</code>	11	D	
<code>\apptocmd</code>	854	<code>\DeclareBoolOption</code>	
<code>\arabic</code>	28 653, 658, 660, 662, 664, 666	
<code>article</code>	11, 15, 43	<code>\DeclareComplementaryOption</code> ..	
<code>article (pkg. opt.)</code>	5, 6 654, 659, 661, 663, 665, 667	
B			
<code>\baselineskip</code>	862	<code>\DeclareDefaultOption</code>	678
<code>\baselinestretch</code>	295	<code>\DeclareStringOption</code>	
<code>bbm</code>	11 650, 651, 652, 655, 656, 657	
<code>\begin</code>	70, 71, 121, 169	<code>\DeclareUrlCommand</code>	471, 792
<code>\beginingroup</code>	66,	<code>default (pkg. opt.)</code>	6, 12
101, 114, 167, 506, 530, 561,		<code>\define@key</code>	668
614, 824, 865, 869, 871, 896, 899		<code>\definecolor</code>	469, 790
<code>\bfseries</code>	13, 151,	<code>\detokenize</code> 57, 80, 85, 132, 640, 816, 883	
551, 553, 728, 759, 918, 926, 934		<code>\dimexpr</code>	556, 862
<code>\bibliography</code>	18, 527, 544	<code>\do</code>	5, 604, 623
<code>bibliographydefs (pkg. opt.)</code> ...	18, 20	<code>dsfont</code>	11
<code>\bibliographystyle</code>	18, 528, 543	E	
<code>bigmargin (pkg. opt.)</code>	12	<code>\email</code>	17, 467, 803
<code>\blacksquare</code>	382	<code>\emergencystretch</code>	308, 317
<code>blockpar (pkg. opt.)</code>	14	<code>\emph</code>	106
C			
<code>\c@footnote</code>	59	<code>\end</code>	92, 93, 123, 174
<code>\c@IndexColumns</code>	779	<code>\endabstract</code>	149, 193, 207
<code>caption</code>	11	<code>\endcmdoptions</code>	898
<code>captions</code>	32	<code>\endcsname</code> 6, 7, 34, 134, 145, 281, 282,	
<code>\captionsetup</code>	298	335, 336, 380, 612, 616, 625,	
<code>\catcode</code>	495, 496,	631, 632, 669, 670, 831, 836, 839	
497, 498, 499, 500, 501, 502, 534		<code>\endgroup</code>	97,
<code>\centering</code>	115, 160, 168	109, 127, 176, 507, 537, 570,	
<code>\changed</code>	815	617, 826, 865, 869, 892, 898, 900	
		<code>\endnoteabstract</code>	207
		<code>\endnotedefaultabstract</code> ..	149, 193
		<code>\endpkgoptions</code>	898
		<code>\endtheglossary</code>	783

144, 148, 149, 192, 193, 206,	
207, 378, 388, 404, 483, 527,	
528, 543, 544, 547, 605, 608,	
612, 782, 783, 786, 803, 824,	
854, 865, 868, 874, 875, 876,	
877, 878, 879, 880, 903, 904, 905	
\lineskip	120
\lowercase	872
ltxdoc	50
M	
\makebox	105, 381, 390
\marg	875
mathtools	11
\meta	863, 865, 868, 874
\metatruefalsearg	863
microtype	11
\mid	863
minimal (pkg. opt.)	11
MinionPro	4
mmynote (pkg. opt.)	4
MyriadPro	4
N	
narrow (pkg. opt.)	10, 12
\newcounter	814
\newlist	852
\newtcolorbox	913, 921, 929
\next	5, 6, 7, 9, 623, 624, 626
nobibliographydefs (pkg. opt.) ...	18
\noexpand	626, 642
nofontdefs (pkg. opt.)	16
nofootnotedefs (pkg. opt.)	17
nohyperrefdefs (pkg. opt.)	17
\noindent	45, 76, 168
noname (pkg. opt.)	10
none (pkg. opt.)	11
\nopagebreak	407
nopagegeomdefs (pkg. opt.)	12
\normalfont	51, 400,
401, 549, 551, 553, 645, 865, 869	
\normalsize	
. 345, 347, 348, 370, 373, 400, 401	
nospacingdefs (pkg. opt.)	16
\noteabstract	206
noteabstract (environment)	164
\noteabstract@attr@compact ...	198
\noteabstract@attr@it	202
\noteabstract@attr@narrow	182
\noteabstract@attr@noname	188
\noteabstract@attr@original ..	191
\noteabstract@attr@small	195
\noteabstract@attr@wide	179
\noteabstract@nameline 159, 170, 189	
\noteabstractafterspacing	
..... 10, 150, 177, 199	
\noteabstractbeforepacing	150
\noteabstractbeforespacing ...	
..... 10, 158, 166, 200	
\noteabstractnamefont . 10, 150, 160	
\noteabstracttextfont	
..... 10, 150, 171, 196, 203	
\noteabstracttextwidth	
..... 10, 150, 169, 180, 185	
\notedefaultabstract	148, 192
notedefaultabstract (environ-	
ment)	148
\noteparagraphfont	
..... 15, 344, 355, 366, 400	
\noteparagraphsetfonts	15, 349
\notesectionallfont	
.. 14, 341, 360, 361, 362, 366, 367	
\notesectionallfontfamily . 14,	
341, 394, 397, 704, 727, 737, 758	
\notesectionfont ... 15, 344, 350, 360	
\notesectionsetfonts 15, 349, 370, 373	
\notesmaller	19, 382, 555,
573, 577, 581, 585, 589, 593, 597	
\notesmaller[0.8]	19
\notesmallerfrac	19, 555, 559
\notesubparagraphfont	
..... 15, 344, 356, 367, 401	
\notesubsectionfont 15, 344, 351, 361	
\notesubsubsectionfont	
..... 15, 344, 352, 362	
\notetitle@style@article	112
\notetitle@style@default	65
\notetitle@style@small	100
\notetitle@thanks	55, 143
\notetitle@thanksmark	53, 144
\notetitle@title	19, 142
\notetitleauthorfont . 7, 13, 82, 105	
\notetitlebelowspace . 7, 16, 98, 110	
\notetitledatefont ... 7, 13, 87, 106	
\notetitlefont 7, 13, 78, 104, 728, 759	
\notetitlehrule	7, 18, 95, 107
\notetitletopspace ... 7, 16, 68, 103	
O	
\oarg	876
opensans	4
\opensans@scale	763

opensansnote (pkg. opt.) 4
 \or 217, 219, 231, 233, 245, 247, 259,
 261, 273, 275, 312, 314, 948, 949
 original (pkg. opt.) 10, 15

P

package options:
 abstract 10, 20
 article 5, 6
 bibliographydefs 18, 20
 bigmargin 12
 blockpar 14
 compact 10, 13
 default 6, 12
 extended 11
 false 12, 16–18
 fontdefs 16, 20
 footnotedefs 16, 17, 20
 hyperrefdefs 17, 20
 indent 15
 indentminiskip 15
 it 10
 itpar 14
 larger 13
 minimal 11
 mmmynote 4
 narrow 10, 12
 nobibliographydefs 18
 nofontdefs 16
 nofootnotedefs 17
 nohyperrefdefs 17
 noname 10
 none 11
 nopagegeomdefs 12
 nospacingdefs 16
 opensansnote 4
 original 10, 15
 pagegeom 12, 20
 pagegeomdefs 12, 20
 par 15, 20
 paragraph 13
 pkgdoc 4
 pkgset 11, 20
 preset 4, 5, 19, 20
 reset 5, 20
 rich 11
 rmfamily 14
 secfmt 13, 14, 20
 secnummargin 14
 secsquares 14
 section 13

section,sffamily 13
 sffamily 13, 14
 sfnote 4
 sfssnote 4
 skip 15
 small 6, 10
 spacingdefs 16, 20
 title 6, 7, 19
 true 12, 16–18
 utopianote 4
 wide 10, 12
 xpkgdoc 4
 xwide 12
 \PackageError .. 22, 137, 634, 672, 760
 \packageoptionsname .. 886, 895, 904
 packages:
 amsfonts 11
 amsmath 11
 amssymb 11
 amsthm 11
 article 11, 15, 43
 bbm 11
 caption 11
 captions 32
 color 38
 csquotes 11
 dsfont 11
 enumitem 11, 52
 etoolbox 21
 float 11
 fontenc 11
 fourier 4, 49
 geometry 12, 30
 graphicx 11
 hyperdoc 50
 hyperref 17, 18, 20, 37, 38, 50
 inputenc 11
 keyval 21
 kvoptions 21, 45
 ltxdoc 50
 mathtools 11
 microtype 11
 MinionPro 4
 MyriadPro 4
 opensans 4
 phfnote ... 1–3, 6, 7, 9–13, 16–19, 30
 phfqitlx 1, 5
 relsize 19
 sectsty 13, 14, 33
 setspace 11, 22, 32

tcolorbox	5, 54	\phfnote@do@secfmt	410, 973
titlesec	13, 33	\phfnote@do@secfmt@blockpar	403
url	17, 38	\phfnote@do@secfmt@compact	369
verbatim	54	\phfnote@do@secfmt@itpar	399
verbdef	11, 49	\phfnote@do@secfmt@larger	372
xcolor	11, 17, 38	\phfnote@do@secfmt@paragraph	364
xkeyval	21	\phfnote@do@secfmt@rmfamily	393
\PackageWarning	9, 284, 300, 338, 490, 857, 860	\phfnote@do@secfmt@secnummargin	387
pagegeom (pkg. opt.)	12, 20	\phfnote@do@secfmt@secsquares	376
pagegeomdefs (pkg. opt.)	12, 20	\phfnote@do@secfmt@section	358
par (pkg. opt.)	15, 20	\phfnote@do@secfmt@sffamily	396
\paragraph	404, 405	\phfnote@do@spacing	289, 980
paragraph (pkg. opt.)	13	\phfnote@docaptionstretch	292, 295, 298, 300
\paragraphfont	366	\phfnote@dostretch	291, 294, 307, 311, 313, 315
\parg	877	\phfnote@email	483, 494, 803
\parindent	44, 72, 323, 327, 331	\phfnote@finalizempfootnotes	42, 91
\parshape	44, 75, 81, 86	\phfnote@fmt@titlefootnotes	47, 50
\parskip	44, 72, 75, 160, 324, 328, 332, 565, 862	\phfnote@format@url	471, 494, 792
\PassOptionsToPackage	212, 216, 218, 220, 226, 230, 232, 234, 240, 244, 246, 248, 254, 258, 260, 262, 268, 272, 274, 276, 436, 441, 444, 456, 472, 519, 709, 716, 718, 724, 735, 740, 742, 755	\phfnote@hook@atendload	681, 693, 738, 789, 1000
phfnote	1–3, 6, 7, 9–13, 16–19, 30	\phfnote@impl@email@	506, 507
\phfnote@bibfont	525, 531	\phfnote@internal@execattribs	4
\phfnote@bibliography	527, 544	\phfnote@internal@execattribs	4, 208, 411, 465
\phfnote@bibliographystyle	539, 543	\phfnote@mpfootmaterial	31, 37, 48
\phfnote@bibstyle	525, 532, 540	\phfnote@mpfootnoteglue	36, 50
\phfnote@changed@set	821, 830	\phfnote@mypfootnotemark	35, 40, 50
\phfnote@changedref@text@par	824, 833, 835	\phfnote@old@bibliography	527, 536
\phfnote@do@bibliographydefs	542, 994	\phfnote@old@bibliographystyle	528, 532
\phfnote@do@fontdefs	517, 990	\phfnote@old@par	73, 74
\phfnote@do@footnotedefs	546, 998	\phfnote@old@paragraph	404, 406
\phfnote@do@noteabstract	205, 971	\phfnote@old@section	608, 609
\phfnote@do@notetitle	131, 969	\phfnote@old@thanks	54
\phfnote@do@pagegeomdefs	280, 976	\phfnote@opt@abstract	752, 959, 971
\phfnote@do@par	334, 983	\phfnote@opt@bibliographydefsfalse	963
\phfnote@do@pdfhyperrefdefs	467, 986	\phfnote@opt@fontdefsfalse	708, 715, 723, 734, 748, 961
\phfnote@do@pkgset	464, 967	\phfnote@opt@fontdefstrue	702
\phfnote@do@pkgset@extended	413	\phfnote@opt@footnotedefsfalse	733, 964
\phfnote@do@pkgset@minimal	413	\phfnote@opt@footnotedefstrue	701
\phfnote@do@pkgset@none	413	\phfnote@opt@hyperrefdefsfalse	788, 960
\phfnote@do@pkgset@rich	413		

<code>\phfnote@opt@pagegeom</code>	<code>\phfnote@url@tilde</code> ...	600, 601, 604
..... 685, 691, 751, 976	<code>\phfnote@urlstyle@common</code> .	574,
<code>\phfnote@opt@pagegeomdefsfalse</code>	578, 582, 586, 590, 594, 598, <u>600</u>	
..... 956	<code>\phfnote@xpkgdoc@old@endtheglossary</code>	783
<code>\phfnote@opt@pagegeomdefstrue</code> 690 783	
<code>\phfnote@opt@par</code> .. 684, 689, 958, 983	<code>\phfnote@xpkgdoc@old@theglossary</code>	782
<code>\phfnote@opt@pkgset</code> 782	
954, 967	<code>\phfnotechanged@tmpa</code>	820, 822
<code>\phfnote@opt@secfmt</code> .. 750, 962, 973	<code>\phfnoteHackSectionStarWithTOC</code> 19, 533, <u>607</u> , 615
<code>\phfnote@opt@spacingdefsfalse</code> 957	<code>\phfnoteHackSectionStarWithTOCInCommand</code> <u>611</u> , 771, 774
<code>\phfnote@opt@title</code> 683, 688, 955, 969	<code>\phfnotePdfLinkColor</code>	17, <u>486</u>
<code>\phfnote@orig@makefnmark</code>	<code>\phfnoteRestoreDefs</code> ...	19, <u>630</u> , 937
547	<code>\phfnoteSaveDefs</code>	19, <u>620</u> , 910
<code>\phfnote@pagegeomstyle@bigmargin</code>	<code>phfquitx</code>	1, 5
..... <u>266</u>	<code>\phfquitlxPkgTitle</code>	938
<code>\phfnote@pagegeomstyle@default</code>	<code>\phfverb</code>	638, 642, 863
..... <u>210</u>	<code>\phfverbfmt</code>	640, 645
<code>\phfnote@pagegeomstyle@narrow</code> <u>224</u>	<code>pkgdoc (pkg. opt.)</code>	4
<code>\phfnote@pagegeomstyle@wide</code> .. <u>238</u>	<code>\pkgfmtdate</code>	943
<code>\phfnote@pagegeomstyle@xwide</code> . <u>252</u>	<code>\pkgfmtdate@next</code>	945, 947, 951
<code>\phfnote@par@indent</code>	<code>\pkgfmtdate@thedata</code>	944, 945
322	<code>\pkgname</code>	806, 811, 938, 939, 941
<code>\phfnote@par@indentminiskip</code> .. 326	<code>\pkgnamefmt</code> ...	807, 808, 809, 812, 813
<code>\phfnote@par@original</code>	<code>\pkgopt@save@meta</code>	868, 869
320	<code>\pkgoptattrib</code>	878, 907
<code>\phfnote@par@skip</code>	<code>\pkgoptattribempty</code>	880, 909
330	<code>\pkgoptattribnodots</code>	879, 908
<code>\phfnote@pkgdoc@index</code> 866, 870	<code>\pkgoptfmt@meta</code>	865, 869
<code>\phfnote@pkgdoc@setupfont</code> 749, 754	<code>\pkgoptionfmt</code>	853, 864, 900
<code>\phfnote@pkgdoc@setupmainfont</code>	<code>\pkgoptions</code>	854, 861, 897
..... 757, 766	<code>\pkgoptions@item@</code>	855, 856
<code>\phfnote@preset@article</code>	<code>\pkgoptions@item@@</code>	855, 857
<u>682</u>	<code>\pkgoptions@old@item</code> .	854, 856, 859
<code>\phfnote@preset@mmynote</code>	<code>\pkgoptionscombineitem</code>	862
<u>699</u>	<code>\pkgoptname</code>	884, 894, 903
<code>\phfnote@preset@opensansnote</code> .	<code>pkgset (pkg. opt.)</code>	11, 20
<u>699</u>	<code>preset (pkg. opt.)</code>	4, 5, 19, 20
<code>\phfnote@preset@pkgdoc</code> ... <u>746</u> , 768	<code>\PrintChanges</code>	770, 771, 777
<code>\phfnote@preset@reset</code>	<code>\PrintChangesAndIndex</code>	777
<u>953</u>	<code>\PrintChangesAndIndexSpacing</code> 776, 777, 778
<code>\phfnote@preset@sfnote</code>	<code>\PrintIndex</code>	773, 774, 778
<u>699</u>	<code>\ProcessKeyvalOptions</code>	966
<code>\phfnote@preset@sfssnote</code>	<code>\protect</code> .. 35, 36, 59, 62, 390, 490, 639	
<u>699</u>	<code>\protected@edef</code>	33, 35, 820
<code>\phfnote@preset@utopianote</code> ... <u>699</u>	<code>\protected@xdef</code>	59, 62
<code>\phfnote@preset@xpkgdoc</code>	<code>\providecommand</code> ...	505, 535, 754, 766
<u>767</u>		
<code>\phfnote@preset@common@xnote</code> ..		
..... <u>687</u> , 700, 722, 732, 747		
<code>\phfnote@requirecolorpackage</code> .		
..... <u>468</u> , <u>508</u>		
<code>\phfnote@sanitize@url</code>		
<u>494</u>		
<code>\phfnote@secnummargin@old@secntformat</code>		
..... 388, 390		
<code>\phfnote@secsquares@old@secntformat</code>		
..... 378, 385		
<code>\phfnote@setupthankssmpfootnote</code>		
..... <u>27</u> , 67		
<code>\phfnote@thanks</code>		
55, 56		
<code>\phfnote@title@checksetspace</code> .		
..... <u>20</u> , 69		
<code>\phfnote@tmp@titsty</code> .. 135, 140, 145		

R	
<code>\raggedright</code>	66, 101
<code>\raise</code>	601
<code>\raisebox</code>	381
<code>\refstepcounter</code>	823
<code>resize</code>	19
<code>\removelastskip</code>	98, 110, 165
<code>\renewenvironment</code>	784
<code>\RequirePackage</code>	
.... 1, 2, 3, 287, 359, 365, 377,	
418, 419, 420, 421, 423, 431,	
432, 434, 437, 439, 442, 445,	
452, 454, 457, 459, 460, 461,	
470, 473, 513, 520, 710, 717,	
719, 725, 726, 736, 739, 741,	
743, 756, 766, 769, 791, 793, 851	
<code>reset</code> (pkg. opt.)	5, 20
<code>\reset@font</code>	46
<code>rich</code> (pkg. opt.)	11
<code>\rmdefault</code>	394
<code>\rmfamily</code>	589
<code>rmfamily</code> (pkg. opt.)	14
<code>\robustify</code>	811, 813, 951
S	
<code>\scriptsize</code>	553
<code>secfmt</code> (pkg. opt.)	13, 14, 20
<code>secnummargin</code> (pkg. opt.)	14
<code>secsquares</code> (pkg. opt.)	14
<code>\section</code>	380, 608, 609
<code>section</code> (pkg. opt.)	13
<code>section,sffamily</code> (pkg. opt.)	13
<code>\sectionfont</code>	360
<code>sectsty</code>	13, 14, 33
<code>\selectfont</code>	105,
342, 526, 549, 557, 581, 600, 602	
<code>\setbox</code>	43
<code>\setcmdnotpkgoptions</code> .	896, 899, 903
<code>\setcounter</code>	780
<code>\setlength</code>	692
<code>\setlist</code>	853
<code>setspace</code>	11, 22, 32
<code>\setstretch</code>	292
<code>\SetupKeyvalOptions</code>	646
<code>\sfdefault</code>	397,
522, 703, 704, 711, 729, 737, 764	
<code>\sffamily</code>	13, 577, 585
<code>sffamily</code> (pkg. opt.)	13, 14
<code>sfnote</code> (pkg. opt.)	4
<code>sfssnote</code> (pkg. opt.)	4
<code>\singlespace</code>	21
<code>skip</code> (pkg. opt.)	15
<code>\small</code>	151, 196, 346, 370
<code>small</code> (pkg. opt.)	6, 10
<code>\space</code>	949, 950
<code>spacingdefs</code> (pkg. opt.)	16, 20
<code>\string</code>	612,
616, 634, 821, 858, 861, 884, 886	
<code>\subparagraphfont</code>	367
<code>\subsectionfont</code>	361
<code>\subsubsectionfont</code>	362
T	
<code>tcolorbox</code>	5, 54
<code>\text</code>	906
<code>\textasciitilde</code>	600, 602
<code>\textbf</code>	843
<code>\textit</code>	849
<code>\textsf</code>	812
<code>\texttt</code>	505, 645
<code>\textwidth</code> .	44, 70, 75, 81, 86, 180, 185
<code>\thanks</code> ...	8, 30, 54, 102, 116, 143, 938
<code>\thanksmark</code>	8, 40, 144
<code>\the</code>	59
<code>\theglossary</code>	782
<code>\thempfootnote</code>	28
<code>\tiny</code>	549, 551
<code>\title</code>	142
<code>title</code> (pkg. opt.)	6, 7, 19
<code>titlesec</code>	13, 33
<code>true</code> (pkg. opt.)	12, 16–18
<code>\ttfamily</code>	573, 865
<code>two-column</code>	3
U	
<code>\underline</code>	906
<code>\unexpanded</code>	381, 642
<code>\unskip</code>	640
<code>url</code>	17, 38
<code>\url</code>	17, 467, 942
<code>\Url@force@Tilde</code>	605
<code>\url@noteitsfstyle</code>	572
<code>\url@noteitstyle</code>	572
<code>\url@notermstyle</code>	572
<code>\url@notesfssstyle</code>	572
<code>\url@notesfstyle</code>	572
<code>\url@notesmlstyle</code>	572
<code>\url@notettstyle</code>	572
<code>\UrlFont</code> 573, 577, 581, 585, 589, 593, 597	
<code>\urlstyle</code>	484, 694, 695, 804
<code>\UrlTildeSpecial</code>	604, 605
<code>\usepackage</code>	912

utopianote (pkg. opt.) 4 \write 821

V

\verb 884, 886
verbatim 54
verbdef 11, 49

W

wide (pkg. opt.) 10, 12

X

\x 872, 873
\xappto 625
xcolor 11, 17, 38
xkeyval 21
xpkgdoc (pkg. opt.) 4
xwide (pkg. opt.) 12