The errata2 package

Manual for version 1.0 (December 18, 2017)

https://github.com/bendikro/LaTeX-errata

Bendik Rønning Opstad
bro.devel+errata@gmail.com

Errata2 package is based on the errata package by Michael Kohlhase

Abstract

This package provides a simple infrastructure for recording errata in LaTeX documents. This allows to maintain an updated version of the document (with all errors corrected) and automatically generate an errata document highlighting the difference to the published version.

Contents

Contents 1

1 Introduction **1**

2 Package Options **1**
2.1 Default command arguments 3

3 Errata commands **3**
3.1 Errata example 4
3.2 Errata environment 4

4 Command Options **5**

5 Errata metadata **6**
5.1 Data keys 6
5.2 Formatter keys 6
5.3 Setting the format for footnotes and errata list 7
5.4 Formatting the keys 7

6 Generating Statistics and Errata Documents **8**

7 Limitations **10**
7.1 Support for special environments 10
7.2 Floating environments 11

8 The Implementation **13**

9 The Errata of this Document **34**

1 Introduction

The life-cycle of a document does not end with its publication. After that, errors will be discovered, and have to be managed. The best way to do this is by marking errata in the text and generating the errata document from that.

2 Package Options

The `errata` package can be called with a variety of package options.

show=⟨*true/false*⟩ (boolean, initially false)

Governs the general visibility of the errata markup. This option enables the options `foots`, `margins`, `inlinechanges`, and `inlinemarks`.

hide=⟨*true/false*⟩ (complementary option to `show`)

Will do the opposite of `show`, ensuring that the errata markup is totally invisible in the formatted document.

margins=⟨*true/false*⟩ (boolean, initially false)

Enables decorations in the margin for each erratum. Setting margins=false will globally disable all margin notes.

foots=⟨*true/false*⟩ (boolean, initially false)

Enables footnote decorations for each erratum.

inlinemarks=⟨*true/false*⟩ (boolean, initially true)

By default, each errata change is marked in the document with the correction encapsulated by paranthesis as shown in example 1. By setting inlinemarks=false, the inline marks will not be added.

inlinechanges=⟨*true/false*⟩ (boolean, initially true)

The default behavior is to make the errata changes visible in the document inlined in the text. By setting inlinechanges=false, the document text will stay unchanged.

record=⟨*true/false*⟩ (boolean, default true)

The errata entries will written to a file named ⟨*jobname*⟩`-errata.tex` so that they can be included in an errata document (see section 6).

hyperref=⟨*true/false*⟩ (boolean, initially true)

This option is used to control whether the `hyperref` package will be used to create hyperlinks between each erratum, the margin notes and the errata list.

cleveref=⟨*true/false*⟩ (boolean, initially true)

This package option will require the `cleveref` package, and changes how labels are created so that the registered errata corrections can be referenced with `cleveref` commands[1].

[1] See `Cleveref support` → P. 23 for the code definition.

marginnote=⟨*true/false*⟩ (boolean, initially true)

The default behavior is to use the \marginpar macro to create the errata reference in the margins. In some environments, such as floating environments, the \marginpar macro cannot not be used. By default, \marginnote from the marginnote package will be used to handle margin notes when necessary.

linenos=⟨*true/false*⟩ (boolean, initially false)

This experimental package option utlizes the lineno package to get the linenumber on the page where the erraum was registered.

styles=⟨*style name*⟩ (string, initially legacy)

The legacy styling of the errata markup in the errata list and the footnotes is very simple. This package option can be used to specify one or more predefined styles. Available styles are: legacy, detailed, tablelist.

requirecmddescription=⟨*true/false*⟩ (boolean, initially true)

By setting this package option to false, the errata command macros (\erratumAdd → P. 3, \erratumDelete → P. 3, \erratumReplace → P. 4 and \erratumNote → P. 4) are defined without the mandatory description argument, i.e. \erratumAdd[⟨*keys*⟩]{⟨*add*⟩}

2.1 Default command arguments

The following package options can be used to specify a set of default settings to be passed to the errata commands described in section 5. The different settings are described in section 4.

The default arguments may be changed using the following package options:

defaultargs={⟨*command options*⟩} (key/values, default described below)

This option allows setting a set of the default arguments to be passed to the errata commands \erratumAdd, \erratumReplace→ P. 4, \erratumDelete, and \erratumNote→ P. 4.

The default values passed to all the commands are as follows[2]:
margin=true
foot=true
inlinemark=true
inlinechange=true
record=true
marginargs={}

defaultargsadd={⟨*command options*⟩} (key/values, initially empty)

This option allows setting a set of the default arguments to be passed to the \erratumAdd command. Any arguments set will override the respective argument set with defaultargs.

defaultargsdelete={⟨*command options*⟩} (key/values, initially empty)

This option allows setting a set of the default arguments to be passed to the \erratumDelete command. Any arguments set will override the respective argument set with defaultargs.

defaultargsreplace={⟨*command options*⟩} (key/values, initially empty)

This option allows setting a set of the default arguments to be passed to the \erratumReplace→ P. 4 command. Any arguments set will override the respective argument set with defaultargs.

defaultargsnote={⟨*command options*⟩} (key/values, initially empty)

This option allows setting a set of the default arguments to be passed to the \erratumNote→ P. 4 command. Any arguments set will override the respective argument set with defaultargs.

3 Errata commands

Marking errata changes in the LaTeX text is done using a few simple macros.

\erratumAdd[⟨*keys*⟩]{⟨*desc*⟩}{⟨*add*⟩}

marks up a correction by adding the text ⟨*add*⟩ for addition with ⟨*desc*⟩ describing the intended correction.

\erratumDelete[⟨*keys*⟩]{⟨*desc*⟩}{⟨*delete*⟩}

marks up a correction by deleting the text ⟨*delete*⟩ with ⟨*desc*⟩ describing the intended correction. Not erratumAdd: \erratumAdd

[2] The code defining the default settings are defined in Default command arguments→ P. 16.

`\erratumReplace[`⟨*keys*⟩`]{`⟨*desc*⟩`}{`⟨*old*⟩`}{`⟨*new*⟩`}`

marks up a correction by replacing the text ⟨*old*⟩ with ⟨*new*⟩.

`\erratumNote[`⟨*keys*⟩`]{`⟨*desc*⟩`}{`⟨*note*⟩`}`

creates a note entry that cannot be described by the other commands.

3.1 Errata example

Example 1 shows all editing operations in action.

Example

```
Here we have \erratumAdd{forgotten word}{three} errata in one \erratumDelete{
superfluous phrase}{darned} long \erratumReplace{translated}{Zeile}{line}.
```

Example 1: Some short errata

All of these macros mark the location of the errata in the margin and document the changes in footnote-like structures. The text in example 1 is rendered as follows:

Err(1) Err(2) Err(3)

Here we have [three]$^{1}_{a}$ errata in one []$^{2}_{d}$ long [line]$^{3}_{r}$

If the package option `hyperref` → P. 1 is enabled, the margin note will link to the entry in the errata list produced by calling `\PrintErrata` → P. 8.

3.2 Errata environment

`\begin{erratum}[`⟨*keys*⟩`]{`⟨*desc*⟩`}`
⟨*environment content*⟩
`\end{erratum}`

Extended errata group multiple editing operations into a coherent group via the `erratum` environment. The first argument of this environment is an explanation as for the short errata above. The `erratum` environment provides local versions of the editing markup macros, which behave like those, but lack the first (explanation) argument, which is already given in the environment that contains them.

Example

```
\begin{erratum}[date=2006-07-19,reported-by=Michael Kohlhase]{old should be new}
This is a test of a long erratum
\begin{enumerate}
  \item We can replace \eReplace{oldtext}{newtext}
  \item and \eAdd{new text}
  \item and finally delete old text\eDelete{alltogether}
  \end{enumerate}
\end{erratum}
```

[1] Erratum! Addition (forgotten word): three (added)
[2] Erratum! Deletion (superfluous phrase): darned (deleted)
[3] Erratum! Correction (translated): line (original text was: “Zeile”)

Example 2: An extended erratum with local correction markers

BErr(4)

The text in example 2 would be rendered as

this is a test of a long erratum

1. We can replace [newtext]$^{4:1}_{r}$
2. and [new text]$_{a}$
3. and finally delete old text []$^{4:2}_{d}$

EErr(4)

The `erratum` environment should also be used in situations where the error occurs in an environment, where normal TEX/LATEX processing is suspended, e.g. a `verbatim` environment. In this case, we can use it to attach correction information via the environment, but do not use the local change documentations.

4 Command Options

Each of the errata commands take an optional list of comma separated key/values as argument. This can be used to specify per-command options for each errata entry.

margin=⟨*true/false*⟩ (boolean, defaults to true)

Allows to override the default margin value for this erratum.

foot=⟨*true/false*⟩ (boolean, defaults to true)

Allows to override the default foot value for this erratum.

inlinechange=⟨*true/false*⟩ (boolean, defaults to true)

Allows to override the default inlinechange value for this erratum.

inlinemark=⟨*true/false*⟩ (boolean, defaults to true)

Allows to override the default inlinemark value for this erratum.

label=⟨*name of label*⟩ (initially empty)

Allows specifying a custom label name to be registered for this errata entry.

uselabel=⟨*name of label*⟩ (string, defaults to errata<X>)

Allows specifying a label to use instead of the default label generated for each registered erratum entry (errata<X> where X is the erratum number). This is necessary in environments where only one label may exist, such as in the `equation` environment.

marginargs={⟨*key/valoptions*⟩} (list, initially empty)

Allows specifying custom arguments to be passed to the marginnote formatter.

usemarginnote[=⟨*length*⟩] (length, initially 0pt)

Forces the use of `\marginnote` for the errata command. An optional value can be specified which denotes how much to verticaly shift the margin note.

[4] Erratum! old should be new
[4:1] was: oldtext
[4:2] deleted: alltogether

5 Errata metadata

5.1 Data keys

For each errata command, a set of meta-data can be specified for an errata entry. The following keys can be used to registered additional data:

date=⟨*date when registered*⟩ (string, initially empty)

Allows specifying the date when the erratum was registered.

reported-by=⟨*name of reporter*⟩ (string, initially empty)

Allows specifying the name of the reporter.

description=⟨*description*⟩ (string, initially empty)

Allows specifying the description of the change.
If the package option `requirecmddescription` → P. 2 is true (the default), This key defaults to the value specified in the ⟨*desc*⟩ field.

location=⟨*location hint*⟩ (string, initially empty)

Used to specify the location in the text where the change is made.

type=⟨*type of correction*⟩ (string, initially empty)

Used to specify what type of change is registered, e.g. grammar, clarification, layout.

Remember to encapsulate any of values in curly braces if it contains a comma.

5.2 Formatter keys

For each of the metadata keys specified in section 5.1, a formatter is defined for the errata list and the footnotes. The formatter defines a custom formatting rule which will be applied to the input value before being used.

Additionally, the following formatter keys exist which may **not** be passed as argument to the errata commands:

pageref (name of label, errata<X>)

Formats a reference to the page number where the erratum is located.

actiondescription

the formatted description of the action

action

The action name

erratumlistitem

This key is used to insert an `\item[]` in the errata list.

erratumlistitemtitle

The title for the erratum in the errata list

erratumlisttarget

The name used when creating a \label target in the errata list.

marginfmt/marginnote

Formats the marginnote

5.3 Setting the format for footnotes and errata list

The formatting of the footnotes and errata list can be changed using the two macros \erratasetListFormat and \erratasetFootFormat.

The argument ⟨*keys*⟩ should specify a comma separated list of keys as specified in section 5[3].

\erratasetFootFormat{⟨*keys*⟩}

Allows changing how the footnotes are formatted.

```
\erratasetFootFormat{+description, +actiondescription}
```

\erratasetListFormat{⟨*keys*⟩}

The default format for the errata list is defined with:

```
\erratasetListFormat{+erratumlisttarget, +description}
```

5.4 Formatting the keys

The format of each of the keys can be set using the macro \erratasetKeyFormat.

\erratasetKeyFormat{⟨*key*⟩}{⟨*format*⟩}

should be used to set the formatter key for a meta data key.

The default format for the keys are as as follows[4]:

[3] See **Errata default style definition** → P. ?? for the code definition

[4] This code is defined at **Default formatters** → P. ??.

```
%%
\NewDocumentCommand{\erratalistitem}{ m }{\item[#1]}
\erratataset{%
  styles/basekeyformats/.style={%
    actionidformat/add/.setStrKeyFmt={##1}{Addition},
    actionidformat/delete/.setStrKeyFmt={##1}{Deletion},
    actionidformat/note/.setStrKeyFmt={##1}{Note},
    actionidformat/replace/.setStrKeyFmt={##1}{Correction},
    %%
    listformats/listbeforenewgroup/.setStrKeyFmt={##1}{},
    listformats/listafternewgroup/.setStrKeyFmt={##1}{},
    listformats/listbeforesamegroup/.setStrKeyFmt={##1}{},
    listformats/listaftersamegroup/.setStrKeyFmt={##1}{},
    %% The name of the errata labels
    errformats/errfmt/.setStrKeyFmt={##1}{Err(##1)},
    errformats/eerrbeginfmt/.setStrKeyFmt={##1}{BErr(##1)},
    errformats/eerrendfmt/.setStrKeyFmt={##1}{EErr(##1)},
    %% Set the format for the action description for each erratum command
    actionformat/add/.setStrKeyFmt={##1}{ ##1 },
    actionformat/delete/.setStrKeyFmt={##1}{##1},
    actionformat/replace/.setStrKeyFmt={##1+##2}{ ##2 --> ##1},
    %%
    listformats/erratumlabel/.setStrKeyFmt={##1}{\@errata@labelname},
    listformats/erratumlistitem/.setStrKeyFmt={##1}{\erratalistitem{##1}},
    listformats/erratumlisttarget/.setStrKeyFmt={##1}{\@errata@htargeterratum{##1}{}},
    listformats/erratumnamedlink/.setStrKeyFmt={##1+##2}{\@errata@hlink{##1}{##2}},
    %%
    listformats/erratalistbegin/.setStrKeyFmt={##1}{\makeatletter\begin{description}},
      listformats/erratalistend/.setStrKeyFmt={##1}{\end{description}\makeatother},
  },
}
\erratataset{%
  legacykeyformats/.style={%
    %% Formats for the footnotes
    footformats/description/.setStrKeyFmt={##1}{~##1},
    footformats/action/.setStrKeyFmt={##1}{~##1},
    footformats/actiondescription/.setStrKeyFmt={##1}{~##1},
    %% Set the format for the action description for each erratum command
    actionformat/add/.setStrKeyFmt={##1}{ (added text)},
    actionformat/delete/.setStrKeyFmt={##1}{(deleted ``##1'')},
    actionformat/replace/.setStrKeyFmt={##1+##2}{ (original text was: ``##2'')},
    %% Errata list format
    listformats/description/.setStrKeyFmt={##1}{~##1},
    listformats/erratumlistitemtitle/.setStrKeyFmt={##1+##2}{%
      \@errata@hlink{##1}{\protect\makebox[\@errata@groupindent][l]{##2}}},
    listformats/type/.setStrKeyFmt={##1}{Type: ##1 },
    listformats/date/.setStrKeyFmt={##1}{ (Date reported: ##1)},
    listformats/reported-by/.setStrKeyFmt={##1}{Reported by: ##1},
    listformats/action/.setStrKeyFmt={##1}{~##1},
    listformats/actiondescription/.setStrKeyFmt={##1}{~##1},
  },
```

6 Generating Statistics and Errata Documents

\PrintErrata

This command inserts the errata list at the current position.

`\printerrata[`⟨*postfix(-errata.tex)*⟩`]{`⟨*prefix*⟩`}`

This command allows to print the errata for another document. This command is useful when generating errata documents for published works. Say we have a book with a driver file `thebook.tex`, into which we have incorporated errata markup using the infrastructure detailed above. Then we have a new document called e.g. `theerrata.tex` which has the form given in example 3. Note that we have used `\printerrata`→ P. 9`{thebook}` to include the errata notices generated from `thebook.tex`.

`\begin{errata}`
⟨*environment content*⟩
`\end{errata}`

The errata list produced when the `record`→ P. 1 option is enabled is placed inside this environment.

`\erratumItem[`⟨*macro*⟩`]{`⟨*title*⟩`}{`⟨*text*⟩`}`

Errata can be marked up using the `\erratumItem` macro in the `errata` environment. `\erratumItem` takes one optional, and two required arguments.
The first argument (`[`⟨*macro*⟩`]`) can be used to specify a macro containing a value that will be expanded after the other values, i.e. at the end.
The second argument `{`⟨*title*⟩`}` will be formatted using the listformats/ `erratumlistitem`→ P. 6 key formatter. For the default formatting of the errata list this is used to insert a `\item[`⟨*title*⟩`]` before the remaining errata meta data specified in the third argument `{`⟨*text*⟩`}`

Normally it is not necessary to manually use this macro, as the `record`→ P. 1 option will write an `erratumItem` the to the errata for each registered erratum.

`\erratamessage`

Putting the macro `\erratamessage` just before the `\end{document}` will generate a message with cardinality information for the errata into the log file.

In the errata document in example 3 we postulate that we keep an updated version of The Book online[5] using the infrastructure provide by the `errata` package. In the updated version of `thebook.tex`, it can be useful to tabulate the errata as well, e.g. in a section in the appendix. This can be done by the `\PrintErrata`→ P. 8 command. Note that this command needs to close the errata file `thebook-errata.tex` therefore we need a `\newpage` to clear the queue of waiting `\writes` before `thebook-errata.tex` can be loaded (otherwise we may be missing the errata from the last page).

[5] And indeed it is good practice to do so if the copyright agreement with the publisher allows this.

Example

```
\documentclass{article}
\usepackage[hide]{errata2}
\title{Errata for The Book}
\begin{document}
\maketitle
\begin{abstract} This document tracks the errata in The Book. \end{abstract}
\section{Introduction}
The errata of The Book are tracked in this document, whose newest version
can be found at \url{.../berrata.pdf}. A version of The Book that contains
all errata corrections (and markup of what changed) can be found at
\url{.../book.pdf}.

In the following we will tabulate the errata in document order. Their
location will be referenced by the section they appear in rather than the
page number, since we do not expect the former to change in the errata
correction process.

\section{The Errata in The Book}
\printerrata{thebook}
\end{document}
```

Example 3: A Sample Errata Document

7 Limitations

If you want to contribute, please contact the author or make a pull request at https://github.com/KWARC/LaTeX-errata

7.1 Support for special environments

When registering an erratum in environments such as `equation` (from `amsmath` package), labels cannot be created freely, therefore the label already registered in the environment must be passed to the errata command.

Example 4 shows example where the label is specified manually.

Title

```
\begin{equation}\label{customlabel}
\begin{split}
  a=a  \\
  b=b \erratumAdd[uselabel=customlabel]{label}{custom label example}
\end{split}
\end{equation}
```

Err(5)

$$\begin{aligned} a &= a \\ b &= b[customlabelexample]_{a}^{5} \end{aligned} \tag{1}$$

[5] Erratum! Addition (label): custom label example (added)
[5] Erratum! Addition (label): custom label example (added)

Example 4: Some short errata

7.2 Floating environments

When registering errata corrections inside floating environments, some manual adjustments may be required. Example 5 shows how the margins can be adjusted manually.

Title

```
In this sentence\erratumAdd[inlinemark=false,label={comma1}]{added comma}
{,} we added a comma with inline mark
disabled\erratumAdd[inlinemark=true,label={comma2}] {added another
comma}{,} as well as a comma with inlinemarking.

Inside environments where \refAuxcs{marginnote} must be used, it may be
necesssary to manually adjust the vertical placement of the margin note
when multiple notes overlap.

As this code is processed inside a \refAux{tcolorbox}, \refAuxcs{marginnote}
must be used instead of \refAuxcs{marginpar}. As we can see, \Cref{comma1}
and \cref{comma2} are overlapping in the margin as
\refAuxcs{marginnote} does not handle stacking multiple notes. Therefore,
we must manually ask \refAuxcs{marginnote} to shift the second margin
value, such as with \erratumAdd[inlinemark=true] {added missing
word}{these} two \erratumAdd[inlinemark=true,usemarginnote=20pt] {added
another missing word}{corrections}.
```

Err(6) Err(7)

In this sentence, we added a comma with inline mark disabled[,]$^{7}_{a}$ as well as a comma with inlinemarking.
Inside environments where \marginnote must be used, it may be necesssary to manually adjust the vertical placement of the margin note when multiple notes overlap.
As this code is processed inside a tcolorbox, \marginnote must be used instead of \marginpar. As we can see, Err(6) and Err(7) are overlapping in the margin as \marginnote does not handle stacking multiple notes. Therefore, we must manually ask Err(8) \marginnote to shift the second margin value, such as with [these]$^{8}_{a}$ two [corrections]$^{9}_{a}$.

Err(9)

[6] Erratum! Addition (added comma): , (added)
[7] Erratum! Addition (added another comma): , (added)
[8] Erratum! Addition (added missing word): these (added)
[9] Erratum! Addition (added another missing word): corrections (added)

Example 5: Example showing how to adjust the margin notes

8 The Implementation

```
\NeedsTeXFormat{LaTeX2e}[1999/12/01]
\ProvidesPackage{errata2}[2017/12/11 v1.0 Errata Management]
\def\erratatwopackage2version{1.0}
\RequirePackage{xifthen}
\RequirePackage{pgfkeys}
\RequirePackage{pgffor}
\RequirePackage{etoolbox}
\RequirePackage{xpatch}
\RequirePackage{xparse}
\RequirePackage{xinttools} % defines \xintFor macro
\RequirePackage[family=@errata@]{errata2strkeyformatter}
\RequirePackage{kvoptions}[2009/07/21]
```

Setup package options

```
\SetupKeyvalOptions {
  family = Errata,
  prefix = @errataOpt@,
}
```

`show` The `show` package option shows everything

```
\DeclareBoolOption{show}
```

`hide` The `hide` package option hides everything

```
\DeclareComplementaryOption{hide}{show}
```

`margins` The `margins` package option

```
\DeclareBoolOption{margins}
```

`foots` The `foots` package option

```
\DeclareBoolOption{foots}
```

`record` The `record` package option shows everything

```
\DeclareBoolOption[true]{record}
```

`requirecmddescription` The `requirecmddescription` package option

```
\DeclareBoolOption[true]{requirecmddescription}
```

`hyperref` The `hyperref` package option

```
\DeclareBoolOption[true]{hyperref}
```

`cleveref` The `cleveref` package option

```
\DeclareBoolOption[true]{cleveref}
```

`marginnote` The `marginnote` package option

```
\DeclareBoolOption[true]{marginnote}
```

`linenos` The `linenos` package option

```
\DeclareBoolOption[false]{linenos}
```

`environmenttestmode` The `environmenttestmode` package option allows to set when `\marginnnote` is used instead of `\marginpar`. Errata will try to detect when `\marginnnote` should be used instead of `\marginpar`, and this option allows for finetuning how safe the test is. The default (`environmenttestmode=0`) will only use `\marginnnote` when it knows for certain that it cannot use `\marginpar`. This is in `math mode`, in `inner mode`, and in tcolorbox's `tcb@savebox` environment. There are certainly many more that should be added to this list. By setting `environmenttestmode=1`, `\marginnnote` will be used whenever `\@captype` is defined, which should include most floating environments such as table and figure. The strictest mode `environmenttestmode=2` will force `\marginnnote` to be used whenever the current environment is not 'document', which means any environment outside of the root document.

```
\DeclareStringOption[0]{environmenttestmode}
```

`uniquelistlabels` The `uniquelistlabels` package option Disabled by default (0), if option set, default level is 3.

```
\DeclareStringOption[0]{uniquelistlabels}[3]
```

`inlinemarks` The `inlinemarks` package option

```
\DeclareBoolOption[true]{inlinemarks}
```

`inlinechanges` The `inlinechanges` package option

```
\DeclareBoolOption[true]{inlinechanges}
```

`styles` The `styles` package option

```
\DeclareStringOption[legacy]{styles}
```

`defaultargs` The `defaultargs` package option

```
\DeclareStringOption[]{defaultargs}
```

`defaultargsadd` The `defaultargsadd` package option

```
\DeclareStringOption[]{defaultargsadd}
```

`defaultargsdelete` The `defaultargsdelete` package option

```
\DeclareStringOption[]{defaultargsdelete}
```

`defaultargsreplace` The `defaultargsreplace` package option

```
\DeclareStringOption[]{defaultargsreplace}
```

defaultargsnote The `defaultargsnote` package option

```
\DeclareStringOption[]{defaultargsnote}

\ProcessKeyvalOptions{Errata}
%% Utility macros
\input{errata2pgfkeysextra.sty}
\NewDocumentCommand{\@errata@requirepackage}{ m }{%
  \@ifpackageloaded{#1}{}{
    \PackageWarning{errata2}{The current errata settings require the #1 package to be loaded.
      Loading package #1...}
    \RequirePackage{#1}
  }
}
\def\erratataset{\pgfqkeys{/@errata@}}
\DeclareExpandableDocumentCommand{\@errata@setvalue}{ m } {\pgfkeys{/@errata@/#1}}
\DeclareExpandableDocumentCommand{\@errata@setvalues}{ O{} m } {\pgfkeys{/@errata@/#1.cd, #2}}
\DeclareExpandableDocumentCommand{\@errata@getvalue}{ m } {\pgfkeysvalueof{/@errata@/#1}}
\NewDocumentCommand{\@errata@ifnotempty}{ m m }{\ifthenelse{\equal{#1}{}}{}{#2}}
%%\NewDocumentCommand{\@errata@ifelsenotempty}{ m m }{\ifthenelse{\equal{#1}{}}{}{#2}}
\NewDocumentCommand{\@errata@ifempty}{ m m }{\ifthenelse{\equal{#1}{}}{#2}{}}
\NewDocumentCommand{\@errata@ifelseempty}{ m m m }{\ifthenelse{\equal{#1}{}}{#2}{#3}}
\NewDocumentCommand{\@errata@ifstrequal}{ m m m }{\ifthenelse{\equal{#1}{#2}}{#3}{}}
%%
```

Default command arguments

Define the custom default arguments for the errata command

```
\pgfkeys{
  /@errata@/cmdkeys/.is family, /@errata@/cmdkeys,
  defaultargs/.style={record=true, foot=true, margin=true, marginargs={},
    lineno=true, inlinemark=true, inlinechange=true},
  defaultargsadd/.style={},
  defaultargsreplace/.style={},
  defaultargsdelete/.style={},
  defaultargsnote/.style={},
  /handlers/.erratаapplystyles/.code={\xintFor ##9 in {#1} \do {\erratataset{styles/##9}}},
}
\erratataset{customformatkeys/.initial={}}
%% Add any default arguments specified by the user
\@errata@setvalues[cmdkeys/]{defaultargs/.append style/.expand once={{\@erratаOpt@defaultargs}}}
\@errata@setvalues[cmdkeys/]{defaultargsadd/.append style/.expand once={\@erratаOpt@defaultargsadd}}
\@errata@setvalues[cmdkeys/]{defaultargsreplace/.append style/.expand once={\@erratаOpt@defaultargsreplac
\@errata@setvalues[cmdkeys/]{defaultargsdelete/.append style/.expand once={\@erratаOpt@defaultargsdelete}
\@errata@setvalues[cmdkeys/]{defaultargsnote/.append style/.expand once={\@erratаOpt@defaultargsnote}}
%%
\if@erratаOpt@show
\@erratаOpt@marginstrue%
\@erratаOpt@footstrue%
\@erratаOpt@inlinemarkstrue%
```

```
\@errataOpt@inlinechangestrue%
\fi
%%
\@ifpackageloaded{hyperref}{}{\@errataOpt@hyperreffalse }
\@ifpackageloaded{cleveref}{}{\@errataOpt@clevereffalse }
\if@errataOpt@marginnote
\@ifpackageloaded{marginnote}{}{\RequirePackage{marginnote}}
\fi
%%
\newcounter{@errata@erratum}
\newcounter{erratum@note}[@errata@erratum]
\if@errataOpt@record\newwrite\@errata@file
\immediate\openout\@errata@file=\jobname-errata.tex
\AtEndDocument{\closeout\@errata@file}\fi
%%
\newlength{\@errata@groupindent}
\setlength\@errata@groupindent{14.7pt}
\newcounter{@errata@uniquelistlabels}
\setcounter{@errata@uniquelistlabels}{\@errataOpt@uniquelistlabels}
\newcounter{@errata@environmenttestmode}
\setcounter{@errata@environmenttestmode}{\@errataOpt@environmenttestmode}
\NewDocumentCommand{\@errata@declareKey}{ O{UNDEF} m }
{\@errata@setvalue{#2/.initial={#1}}}
%%
\NewDocumentCommand{\@errata@declareStoreKey}{ O{UNDEF} m }
{%% Basically 'estore in', but must build macro name from argument
\pgfkeys{/@errata@/#2/.code={\expandafter\edef\csname @errata@#2\endcsname{##1}}}%
\pgfkeys{/@errata@/#2={#1}}%
}
%%
\NewDocumentCommand{\@errata@setcmdkeys}{ O{0} m }{%
\xintFor ##9 in {#2} \do {%
\edef\@errata@key{##9}%
\@errata@ifnotempty{##9} {%
  \@errata@setvalue{cmdkeys/\@errata@key/.initial=#1}%
}}}
%%
\NewDocumentCommand{\erratumSetFootFormat}{ m }{%
  \@errata@declareStoreKey[#1]{footformat}%
}
\NewDocumentCommand{\erratumSetListFormat}{ m }{%
  \@errata@declareStoreKey[#1]{listformat}%
}
\@errata@declareStoreKey[type, description, location, date, id, pageref, lineref,
inlinemark, inlinechange, uselabel, label, margin, foot, record, lineno,
marginargs, usemarginnote, reported-by, actionid, action, actionargs, actiondescription,
erratumlistitem, erratumlistitemtitle, erratumlisttarget,
listbeforenewgroup, listafternewgroup, listbeforesamegroup, listaftersamegroup,
erratumlabel, erratumname, erratumnamedlink]{formatkeys}% Defines \@errata@formatkeys
%%
```

```
\@errata@setcmdkeys{\@errata@formatkeys}
\@errata@setvalue{cmdkeys/usemarginnote/.default={0pt}}
%%
```

\erratasetkeyformat Macro for setting the format of a key

```
\NewDocumentCommand{\erratasetkeyformat}{ O{} m m }{% PGF argument pattern, key, formated output
  \strkeyfmtsetkeyformat[#1]{#2}{#3}%
}
```

rrataRegisterFormatKey Macro for registering a new format key

```
\NewDocumentCommand{\erratasetkeyformat}{ m }{%
  \erratasetset{customformatkeys/.append=#1}%
  \erratasetset{formatkeys/.append code=#1}%
  \@errata@setcmdkeys{#1}
}

%%
%%      Set default formater for all keys
\SetDefaultKeylistFormat[#1]{listformats}{\@errata@formatkeys}
\SetDefaultKeylistFormat[#1]{footformats}{\@errata@formatkeys}
\SetDefaultKeylistFormat[#1]{errformats}{errfmt, eerrbeginfmt, eerrendfmt}
\SetDefaultKeylistFormat[#1]{actionformat}{add, delete, replace, note}
\erratasetkeyformat[#1+#2]{marginfmt/marginnote}{#2}%
%%
\@errata@declareKey[{@errata@erratum}]{footnoteprefix}
\@errata@declareKey[\scshape{Erratum!}]{footnotebegin}
%%
%% Used internally to track label and label references
\@errata@declareKey{labelref}
\@errata@declareStoreKey{erratumname}
\@errata@setvalue{labelname/.gstore in=\@errata@labelname}
\@errata@setvalue{labelname={}}
\@errata@setvalue{prevlabel/.gestore in=\@errata@prevlabel}
\@errata@setvalue{prevlabel={}}
\@errata@setvalue{lastuniquelabel/.gestore in=\@errata@lastuniquelabel}
\@errata@setvalue{lastuniquelabel={UNDEF}}
\@errata@setvalue{newlabelgroup/.gstore in=\@errata@newlabelgroup}
\@errata@setvalue{newlabelgroup={true}}
%%
\@errata@setvalue{lastcommandinlinevalue/.gstore in=\lastcommandinlinevalue}
\@errata@setvalue{lastcommandinlinevalue={}}
\@errata@setvalue{inlinevalue/.initial={}}
```

Errata legacy style definition

Define the legacy errata formatting

```
%%
\NewDocumentCommand{\erratalistitem}{ m }{\item[#1]}
\erratasetset{%
```

```
    styles/basekeyformats/.style={%
      actionidformat/add/.setStrKeyFmt={##1}{Addition},
      actionidformat/delete/.setStrKeyFmt={##1}{Deletion},
      actionidformat/note/.setStrKeyFmt={##1}{Note},
      actionidformat/replace/.setStrKeyFmt={##1}{Correction},
      %%
      listformats/listbeforenewgroup/.setStrKeyFmt={##1}{},
      listformats/listafternewgroup/.setStrKeyFmt={##1}{},
      listformats/listbeforesamegroup/.setStrKeyFmt={##1}{},
      listformats/listaftersamegroup/.setStrKeyFmt={##1}{},
      %% The name of the errata labels
      errformats/errfmt/.setStrKeyFmt={##1}{Err(##1)},
      errformats/eerrbeginfmt/.setStrKeyFmt={##1}{BErr(##1)},
      errformats/eerrendfmt/.setStrKeyFmt={##1}{EErr(##1)},
      %% Set the format for the action description for each erratum command
      actionformat/add/.setStrKeyFmt={##1}{ ##1 },
      actionformat/delete/.setStrKeyFmt={##1}{##1},
      actionformat/replace/.setStrKeyFmt={##1+##2}{ ##2 --> ##1},
      %%
      listformats/erratumlabel/.setStrKeyFmt={##1}{\@errata@labelname},
      listformats/erratumlistitem/.setStrKeyFmt={##1}{\erratalistitem{##1}},
      listformats/erratumlisttarget/.setStrKeyFmt={##1}{\@errata@htargeterratum{##1}{}},
      listformats/erratumnamedlink/.setStrKeyFmt={##1+##2}{\@errata@hlink{##1}{##2}},
      %%
      listformats/erratalistbegin/.setStrKeyFmt={##1}{\makeatletter\begin{description}},
        listformats/erratalistend/.setStrKeyFmt={##1}{\end{description}\makeatother},
    },
  }
  \erratataset{%
    legacykeyformats/.style={%
      %% Formats for the footnotes
      footformats/description/.setStrKeyFmt={##1}{~##1},
      footformats/action/.setStrKeyFmt={##1}{~##1},
      footformats/actiondescription/.setStrKeyFmt={##1}{~##1},
      %% Set the format for the action description for each erratum command
      actionformat/add/.setStrKeyFmt={##1}{ (added text)},
      actionformat/delete/.setStrKeyFmt={##1}{(deleted ``##1'')},
      actionformat/replace/.setStrKeyFmt={##1+##2}{ (original text was: ``##2'')},
      %% Errata list format
      listformats/description/.setStrKeyFmt={##1}{~##1},
      listformats/erratumlistitemtitle/.setStrKeyFmt={##1+##2}{%
        \@errata@hlink{##1}{\protect\makebox[\@errata@groupindent][l]{##2}}},
      listformats/type/.setStrKeyFmt={##1}{Type: ##1 },
      listformats/date/.setStrKeyFmt={##1}{ (Date reported: ##1)},
      listformats/reported-by/.setStrKeyFmt={##1}{Reported by: ##1},
      listformats/action/.setStrKeyFmt={##1}{~##1},
      listformats/actiondescription/.setStrKeyFmt={##1}{~##1},
    },
    legacylistandfootstyle/.code={%
```

Legacy style formatters

Define the legacy format of the errata list and footnotes

```
    \errataSetListFormat{+erratumlisttarget, +description}
    \errataSetFootFormat{+description, +actiondescription}
  },
  styles/legacy/.style={styles/basekeyformats, legacykeyformats, legacylistandfootstyle},
}
%%
%% END style: legacy
%%
```

Errata detailed style definition

Define the detailed formatting

```
%%
\erratataset{%
  detailedstylekeyformats/.style={%
    %% Errata list format
    listformats/description/.setStrKeyFmt={##1}{~(##1)},
    listformats/erratumlistitemtitle/.setStrKeyFmt={##1+##2}{%
      {\@errata@hlink{##1}{%
          \protect\makebox[\@errata@groupindent][l]{##2}}%
        \@errata@hlinkerratum{\the@errata@erratum}{\@errata@erratumname}}},
    listformats/action/.setStrKeyFmt={##1}{~##1},
    listformats/actiondescription/.setStrKeyFmt={##1}{\newline ##1},
    listformats/pageref/.setStrKeyFmt={##1}{Page~\pageref{##1}},
    listformats/lineref/.setStrKeyFmt={##1}{Line \lineref{##1}},
    %% Foot format
    footformats/description/.setStrKeyFmt={##1}{~(##1)},
    footformats/action/.setStrKeyFmt={##1}{~##1},
    footformats/actiondescription/.setStrKeyFmt={##1}{~##1},
    %% Action format
    actionformat/replace/.setStrKeyFmt={##1+##2}{##1 (original text was: ‘‘##2’’)},
    actionformat/add/.setStrKeyFmt={##1}{##1 (added)},
    actionformat/delete/.setStrKeyFmt={##1}{##1 (deleted)},
  },
  styles/margintcolorboxstyle/.style={
    marginfmt/marginnote/.setStrKeyFmt={##1+##2}{%
      {\vspace{-4pt}%
        {\begin{tcolorbox}[size=fbox,nobeforeafter,
            colframe=red!50!white,colback=red!25!yellow!5!white,%
            leftrule=0pt,rightrule=0pt,bottomrule=1pt,toprule=1pt,
            arc=0cm, outer arc=0mm, % Remove the rounded corners
            fontupper=\scriptsize,%
            code={\pgfkeysalsofrom##1}
            ]##2\end{tcolorbox}}}},%
```

```
  },
  detailedlistandfootstyle/.code={%
    \erratasetListFormat{+erratumlisttarget, +pageref, +action, -description, -date, +actiondescription}
    \erratasetFootFormat{+action, -description,:, +actiondescription}
  },
  detailedstylesetupcode/.code={%
    \newlength{\@errata@listindent}
    \setlength\@errata@listindent{33mm}
    \setlength\@errata@groupindent{16mm}
    \setcounter{@errata@uniquelistlabels}{3}
    \AtBeginDocument{%
      \@errata@requirepackage{tcolorbox}
      \@errata@requirepackage{enumitem}
      \AtBeginEnvironment{errata} {
        \setlist[description]{labelindent=0pt,labelwidth=\@errata@listindent,leftmargin=!}%
      }
    }
  },
  styles/detailed/.style={detailedstylekeyformats, detailedlistandfootstyle,
    detailedstylesetupcode},
}
%%
%% END style: detailed
%%
```

Errata list table style definition

Define the tablelist style for formatting the errata list as a table

```
%%
\NewDocumentCommand{\erratasetamersand}{}{&}
\NewDocumentCommand{\erratalistbeginnewgroup}{}{\\[4pt]\hline\rule[1ex]{0pt}{1em}}
\NewDocumentCommand{\erratacustomendline}{}{\\}
\erratasetset{%
  liststyletablekeyformats/.style={%
    listformats/pageref/.setStrKeyFmt={##1}{\pageref{##1} },
    listformats/description/.setStrKeyFmt={##1}{##1},
    listformats/type/.setStrKeyFmt={##1}{Type: ##1},
    listformats/date/.setStrKeyFmt={##1}{ (Date reported: ##1)},
    listformats/reported-by/.setStrKeyFmt={##1}{Reported by: ##1},
    listformats/actiondescription/.setStrKeyFmt={##1}{ ##1},
    listformats/erratumlistitem/.setStrKeyFmt={##1}{},
    listformats/erratumlistitemtitle/.setStrKeyFmt={##1+##2}{%
      {\@errata@hlink{##1}{\protect\makebox[\@errata@groupindent][l]{##2}}}},
    %%
    listformats/listbeforenewgroup/.setStrKeyFmt={##1}{\noexpand\erratalistbeginnewgroup},
    listformats/listafternewgroup/.setStrKeyFmt={##1}{},
    listformats/listbeforesamegroup/.setStrKeyFmt={##1}{\noexpand\erratacustomendline},
    listformats/listaftersamegroup/.setStrKeyFmt={##1}{},
    %%
```

```
    listformats/erratalistbegin/.setStrKeyFmt={##1}{\makeatletter%
      \begin{longtable}{ p{1.3cm} p{1.3cm} c p{5cm} c }%
        & Title & Type & Description & Page %
        },
    listformats/erratalistend/.setStrKeyFmt={##1}{\end{longtable}\makeatother},
  },
  requirelongtable/.code={%
    \AtBeginDocument{%
      \@errata@requirepackage{longtable}
    }
  },
  liststylefortable/.code={
    \errataSetListFormat{+erratumlistitemtitle,+erratumlisttarget,\noexpand\errataamersand,%
      +erratumnamedlink,\noexpand\errataamersand,+action,\noexpand\errataamersand,%
      +actiondescription,\noexpand\errataamersand,+pageref}%
    \setcounter{@errata@uniquelistlabels}{4}
  },
  styles/tablelist/.style={liststyletablekeyformats, requirelongtable,
    legacylistandfootstyle, liststylefortable},
}
%%
%% END style tablelist
%%
\input{errata2styles.sty}
\errataset{.errataapplystyles/.list={basekeyformats, \@errataOpt@styles}}
%%
\DeclareExpandableDocumentCommand{\@errata@htargetname}{ m } {errata#1}%
\DeclareExpandableDocumentCommand{\@errata@linenoname}{ m } {\if@errataOpt@linenos%
  erratalineno#1\else{}\fi}%
\DeclareExpandableDocumentCommand{\@errata@htargetlisttname}{ m } {erratalist#1}%
\NewDocumentCommand{\@errata@hlink}{ m m } {#2}%
\NewDocumentCommand{\@errata@label}{ m } {%
  \protected@edef\@errata@expanded@arg{#1}\label{\@errata@expanded@arg}}%
\NewDocumentCommand{\@errata@htargeterratum}{ O{} m m } {\@errata@label{#1}#2}
\NewDocumentCommand{\@errata@hlinkerratum}{ m m } {\@errata@hlink{\@errata@htargetname{#1}}{#2}}
\NewDocumentCommand{\@errata@hlinkerratumlist}{ m m } {\@errata@hlink{\@errata@htargetlisttname{#1}}{#2}}
%%
%% Enable internal linking if hyperref
\if@errataOpt@hyperref
\RenewDocumentCommand{\@errata@hlink}{ m m } {%
\protected@edef\@errata@expanded@arg{#1}%
\hyperlink{\@errata@expanded@arg}{#2}}%
%%
%% \@errata@htargeterratum
%% ARG1: Optional argument to indicate whether to add labels,
%% ARG2: The (optional) target name of a custom label
%% ARG3: The target name of the primary label/hyperlink
%% ARG4: The text to insert after the label/hypertarget
%% Hypertarget creates a hyper-link anchor, used to link to this location with \hyperlink
%% The label is needed for such as ref/pageref/cref
```

```
\RenewDocumentCommand{\@errata@htargeterratum}{ O{} O{} m m }{%
 \Hy@raisedlink{\hypertarget{#3}{}}%
  \@errata@ifempty{#1}{% Add labels
    \@errata@label{#3}% Primary label
    \if@erratаOpt@linenos%
    \ifboolexpr{ test {\ifdolineno} }{%
      \edef\@errata@expanded@arg{\@errata@linenoname{\the@errata@erratum}}%
      \ifinner%
      \expandafter\@LN@postlabel\expandafter{\@errata@expanded@arg}%
      \else%
      \expandafter\linelabel\expandafter{\@errata@expanded@arg}%
      \fi%
    }{}%
    \fi%
    \@errata@ifnotempty{#2}{\@errata@label{#2}}% Seconday custom label
  }{#4}%
}%
\fi
%%
```

Cleveref support

Code for cleveref support

```
\if@erratаOpt@cleveref
  \AtBeginDocument{%
    \Crefname{errata}{Err}{Errata}
    \crefname{errata}{err}{errata}
    \crefformat{errata}{Err(#2#1#3)}
    \Crefformat{errata}{Err(#2#1#3)}
  }
  \RenewDocumentCommand{\@errata@label}{ m } {%
    \protected@edef\@errata@expanded@arg{#1}\label[errata]{\@errata@expanded@arg}}%
\fi
%%
\NewDocumentCommand{\@errata@FormatKeywords}{ O{} m m m }{% key, keys/values, format-string
  \@StrKeyFmt@FormatKeywords[#1]{#2}{cmdkeys}{#3}{#4}%
}
%%
\newcounter{@errata@lowerlabellim}
\setcounter{@errata@lowerlabellim}{0}
%% Return true of the label name can be set
\newcommand\ifuniquelabeltest[1]{%
%% Current document class doesn't have chapters.
    \setcounter{@errata@lowerlabellim}{#1}%
  \@ifundefined{thechapter}{%
    \stepcounter{@errata@lowerlabellim}%
  }{}%
  \ifboolexpr{test {\ifnumcomp{\value{@errata@uniquelistlabels}}{=}{0}} or test {\ifnumcomp{\value{@errat
```

```
%%
\DeclareExpandableDocumentCommand{\@errata@setlabelref}{ } {%
\@ifundefined{thechapter}{}{%
  \ifnumcomp{\value{chapter}}{>}{0}{%
    \@errata@setvalue{labelname=\thechapter}%
    \@errata@setvalue{labelref={chapter.\thechapter}}%
  }{}}%
%% Check section
\@ifundefined{thesection}{}{%
  \ifnumcomp{\value{section}}{>}{0}{%
    \@errata@setvalue{labelref={section.\thesection}}%
    \ifboolexpr{test {\ifuniquelabeltest{1}} }{\@errata@setvalue{labelname=\thesection}}{}%
    %% Check subsection
    \@ifundefined{thesubsection}{}{%
      \ifnumcomp{\value{subsection}}{>}{0}{%
        \@errata@setvalue{labelref={subsection.\thesubsection}}%
        \ifboolexpr{test {\ifuniquelabeltest{2}} }{\@errata@setvalue{labelname=\thesubsection}}{}%
        %% Check subsubsection
        \@ifundefined{thesubsubsection}{}{%
          \ifnumcomp{\value{subsubsection}}{>}{0}{%
            \@errata@setvalue{labelref={subsubsection.\thesubsubsection}}%
            \ifboolexpr{test {\ifuniquelabeltest{3}} }{\@errata@setvalue{labelname=\thesubsubsection}}{}%
          }{}%
        }%
      }{}% END subsubsection
    }% END subsection
  }{}%
}% END section
}
%%
```

\erratumItem Macro for adding an item to the errata list

```
%% ARG1: Optional value to add before the content
%% ARG2: Optional value to add after the content
%% ARG3: Item title
%% ARG4: errata-format-text
\DeclareDocumentCommand{\erratumItem}{ O{} O{} m m } {#1#3#4#2}
%%
```

\printerrata Macro for inserting an existing errata list into the current document.

```
\DeclareDocumentCommand{\printerrata}{ O{-errata.tex} m } {%
\IfFileExists{#2#1}{%
\begin{errata}\input{#2#1}\end{errata}}{}%
}
```

\PrintErrata Macro for inserting the errata list produced for the current document.

```
\DeclareDocumentCommand{\PrintErrata}{} {%
\if@erratаOpt@record\immediate\closeout\@errata@file\fi%
\printerrata{\jobname}%
}
```

```
%%
\newcommand\ifdomargin[0]{%
  \ifboolexpr{
    bool{@erratаOpt@margins}
    and test{\ifthenelse{\equal{\@errata@getvalue{cmdkeys/margin}}{true}}}}}
\newcommand\ifdofoot[0]{%
  \ifboolexpr{
    bool{@erratаOpt@foots}
    and test{\ifthenelse{\equal{\@errata@getvalue{cmdkeys/foot}}{true}}}}}
\newcommand\ifdorecord[0]{%
  \ifboolexpr{
    bool{@erratаOpt@record}
    and test{\ifthenelse{\equal{\@errata@getvalue{cmdkeys/record}}{true}}}}}
%%
%% Ideally, the \ifdolineno test would include a test for bool{@erratаOpt@linenos}
%% However, LineNumbers isn't available if the lineno package isn't loaded,
%% and as ifboolexpr doesn't do lazy evaluation, we can't test for that
%% unless we know the package is loaded.
\newcommand\ifdolineno[0]{%
  \ifboolexpr{
    and test{\ifthenelse{\equal{\@errata@getvalue{cmdkeys/lineno}}{true}}}
    and bool{LineNumbers} }}
%%
```

`errata` The errata environment used for the errata list

```
\newenvironment{errata}{%
  \StrKeyFmtgetformatted{listformats/erratalistbegin}{}{\@errata@tmp@erratalistbegin}%
  \@errata@tmp@erratalistbegin%
}{\StrKeyFmtgetformatted{listformats/erratalistend}{}{\@errata@tmp@erratalistend}%
  \@errata@tmp@erratalistend%
}
%%
```

`erratum` The erratum environment used for creating an extended errata.

```
\DeclareDocumentEnvironment{erratum}{ O{} m }% keys, explanation
{\@erratum@cmd{}{#1}{action={Long errata}, description={#2}}%
\expandafter\@errata@htargeterratum{\@errata@htargetname{\the@errata@erratum}}{}%
\if@erratаOpt@foots%
\footnotetext[\value{@errata@erratum}]{{\@errata@getvalue{footnotebegin} #2}}%
\fi%
\protected@edef\@errata@tmp@marginargs{\@errata@getvalue{cmdkeys/marginargs}}%
\ifboolexpr{test {\ifdomargin} }%
{\StrKeyFmtgetformatted{errformats/eerrbeginfmt}{\@errata@hlinkerratumlist{\the@errata@erratum}{\arabic{@
\@errata@marginnote[\@errata@tmp@marginargs]{\@errata@tmpfmtvalue}%
}{}%
\@errata@dorecord{listformats}{\@errata@listformat}{\@errata@argsstring}%
\def\eAdd##1{[##1]$_a$}%
\def\eDelete##1{\erratum@mark[]$_d^{\@thefnmark}$\if@erratаOpt@foots\@footnotetext{deleted: ##1}\fi}%
\def\eReplace##1##2{\erratum@mark[##2]$_r^{\@thefnmark}$\if@erratаOpt@foots\@footnotetext{was: ##1}\fi}
\ignorespaces}%
```

```
{\ifboolexpr{test {\ifdomargin} }%
{\StrKeyFmtgetformatted{errformats/eerrendfmt}{\@errata@hlinkerratumlist{\the@errata@erratum}{\arabic{@er
\@errata@marginnote[\@errata@tmp@marginargs]{\@errata@tmpfmtvalue}%
}{}%
}%

%%
\def\erratum@mark{\stepcounter{erratum@note}{}
\def\@thefnmark{\arabic{@errata@erratum}:\arabic{erratum@note}}}
```

`\erratatemessage` Write info to log file

```
\NewDocumentCommand{\erratamessage}{ m } {% erratakeyvalues
\ifnum\value{@errata@erratum}>0\typeout{}%
\typeout{This document contains \arabic{@errata@erratum} Errata; see \jobname-errata.tex!}%
\typeout{}%
\fi}
```

`\@errata@write` An immediate version of \protected@write to write errata entries to file

```
% get a copy of `\protected@write
\let\@errata@write\protected@write
% patch the copy to add \immediate
\xpatchcmd{\@errata@write}{\write}{\immediate\write}{}{}
```

`\@errata@dorecord`

```
\NewDocumentCommand{\@errata@dorecord}{ m m m } {% formattype, listformat, argsstring
\ifboolexpr{ test {\ifdorecord} }{%
%% Get the formatted code for a new group
\ifboolexpr{ test {\ifthenelse{\equal{\@errata@newlabelgroup}{true}}} }%
{%
  \StrKeyFmtgetformatted{listformats/listbeforenewgroup}{\@errata@labelname}{\@errata@tmp@before}%
  \StrKeyFmtgetformatted{listformats/listafternewgroup}{\@errata@labelname}{\@errata@tmp@after}%
}%
{%
  \StrKeyFmtgetformatted{listformats/listbeforesamegroup}{\@errata@labelname}{\@errata@tmp@before}%
  \StrKeyFmtgetformatted{listformats/listaftersamegroup}{\@errata@labelname}{\@errata@tmp@after}%
}%
\@errata@record[\@errata@tmp@before][\@errata@tmp@after]{\noexpand\@errata@FormatKeywords{#1}{#3}{#2}}%
}{}%
}
```

`\@errata@record` Create an entry in the errata list produced by PrintErrata

```
\NewDocumentCommand{\@errata@record}{ O{} O{} m } {% erratakeyvalues
\@errata@setvalue{cmdkeys/erratumlistitemtitle/.get=\@errata@tmp@itemtitle}%
\StrKeyFmtgetformattedef{listformats/erratumlistitemtitle}{\@errata@tmp@itemtitle}{\@errata@tmp@formatted
\StrKeyFmtgetformatted{listformats/erratumlistitem}{\@errata@tmp@formatteditemtitle}{\@errata@tmplistitem
\@errata@write\@errata@file{}%
{\string\erratumItem[#1][#2]{\@errata@tmplistitem}{#3}}%
}

\newif\if@errata@inbadenvironment
```

```
\newif\if@errata@inknownbadenvironment
%%
\NewDocumentCommand{\@erratum@checkenvironment}{}{%
\@errata@inknownbadenvironmentfalse%
\@errata@inbadenvironmentfalse%
%%
\ifinner%
\@errata@inknownbadenvironmenttrue%
\fi%
\ifmmode%
\@errata@inknownbadenvironmenttrue%
\fi%
\ifthenelse{\equal{\@currenvir}{tcb@savebox}}%
{\@errata@inknownbadenvironmenttrue}%
{}%
%%
\ifnumgreater{\value{@errata@environmenttestmode}}{0}%
{% At least 1, so lets consider any environment with \@captype dangerous
\ifx\@captype\@undefined%
\else%
\@errata@inbadenvironmenttrue%
\fi%
\ifnumequal{\value{@errata@environmenttestmode}}{2}%
{%
\ifthenelse{\equal{\@currenvir}{document}}%
{}{\@errata@inbadenvironmenttrue}%
}{}% End == 2
}{}% End > 0
}
\edef\emptyval{}%
%% Create the margin note
\NewDocumentCommand{\@errata@marginnote}{ O{\emptyval} m }{%
%% We use marginpar if we can, as it handles stacking multiple values vertically
\let\@errata@marginnote@cmd\marginpar%
%% If captype is defined, we're probably inside a float
\if@errata@inknownbadenvironment%
\let\@errata@marginnote@cmd\marginnote%
\fi
\if@errata@inbadenvironment%
\let\@errata@marginnote@cmd\marginnote%
\fi
%% If user provided argument 'usemarginnote', we use \marginnote macro
\protected@edef\@errata@tmp@marginshift{\@errata@getvalue{cmdkeys/usemarginnote}}%
\@errata@ifnotempty{\@errata@tmp@marginshift}{%
  \let\@errata@marginnote@cmd\marginnote%
  \renewcommand*{\marginnotevadjust}{\@errata@tmp@marginshift}%
}%
\StrKeyFmtgetformatted{marginfmt/marginnote}{{#1}+{#2}}{\@errata@tmp@marginfmtvalue}%
\@errata@marginnote@cmd{\@errata@tmp@marginfmtvalue}%
}
```

```
%% Do the pagelayout of the errata (inline changes, margin, and footnote)
\NewDocumentCommand{\@errata@dolayoutonpage}{ m m m m }{% type(a,r,d), original, change, errata-formatted
\ifboolexpr{test {\ifdomargin} }%
{\StrKeyFmtgetformatted{errformats/errfmt}{\@errata@hlinkerratumlist{\the@errata@erratum}{\arabic{@errata
  \protected@edef\@errata@tmp@marginargs{\@errata@getvalue{cmdkeys/marginargs}}%
  \@errata@marginnote[\@errata@tmp@marginargs]{\@errata@tmpfmtvalue}%
}{}%
%% Handle inline changes
\ifboolexpr{ bool{@erratаOpt@inlinechanges} and
  test {\ifthenelse{\equal{\@errata@getvalue{cmdkeys/inlinechange}}{true}}} }%
{% Make inline change
%%\@errata@setvalues{lastcommandinlinevalue={#3}, inlinevalue={#3}}
  \@errata@setvalue{lastcommandinlinevalue={#3}}%
  \@errata@setvalue{inlinevalue={#3}}%
}%
{% Use original value
  \@errata@setvalue{lastcommandinlinevalue={#2}}%
  \@errata@setvalue{inlinevalue={#2}}%
}%
%% If trying to set lastcommandinlinevalue=inlinevalue here with
%% (\@errata@setvalue{lastcommandinlinevalue=\@errata@getvalue{inlinevalue}}%),
%% it adds an unwanted space to lastcommandinlinevalue for some reason. Therefore,
%% both lastcommandinlinevalue and inlinevalue are set to the same value separately.
%%
\ifboolexpr{ bool{@erratаOpt@inlinemarks} and
  test {\ifthenelse{\equal{\@errata@getvalue{cmdkeys/inlinemark}}{true}}} }%
{% Use add inlinemark
  \@errata@setvalue{lastcommandinlinevalue={[]}}%
  \@errata@setvalue{inlinevalue/.prefix={[}}%
  \@errata@setvalue{inlinevalue/.append={]\ensuremath{_#1^{\arabic{@errata@erratum}}}}}}%
}{}%
\protected@edef\@errata@tmp@dolabel{}%
\ifmmode% In math mode, we can't add a label
\eappto\@errata@tmp@dolabel{nolabel}%
\fi%
\@errata@htargeterratum[\@errata@tmp@dolabel][\@errata@getvalue{cmdkeys/label}]%
{\@errata@htargetname{\the@errata@erratum}}{\@errata@getvalue{inlinevalue}}%
%%
\ifboolexpr{test {\ifdofoot} }{%
  \protected@edef\@errata@tmp@footprefix{\arabic{\@errata@getvalue{footnoteprefix}}}% Expand the formatti
  \expandafter\footnotetext\expandafter[\@errata@tmp@footprefix]%
  {{\@errata@getvalue{footnotebegin}}%
    \@errata@ifnotempty{#4}{#4}}%
}{}%
}%
%%
\erratаset{%
  setcmdkeyvalue/.code={%
    %% Apply the registered formatter
    \erratаset{#1/.cd, .@formatter}%
```

```
    %% Get the formatted output
    \erratasetset{#1/.@formatter/.@formatted/.get=\@errata@tmp@cmdkeyargs}%
    %% Update the cmdkey with the formatted output
    \@errata@setvalue{#1/.expand once=\@errata@tmp@cmdkeyargs}%
  },
  setcmdkeyvalues/.style={setcmdkeyvalue=cmdkeys/#1},
  setCmdKeyArgsGetTwo/.style n args={3}{%
    cmdkeys/#1/.setStrKeyWithDefFmt={#2}{#3}{%
      {{\@errata@getvalue{cmdkeys/##1}}+{\@errata@getvalue{cmdkeys/##2}}}}%
  },
  setCmdKeyArgsGetOne/.style n args={3}{%
    cmdkeys/#1/.setStrKeyWithDefFmt={#2}{#3}{{\@errata@getvalue{cmdkeys/##1}}},%
  },
}
%%
\erratasetset{%
  setCmdKeyArgsGetOne={pageref}{#1}{{uselabel}},
  cmdkeys/lineref/.setStrKeyWithDefFmt={#1}{}{\@errata@linenoname{\the@errata@erratum}},
  cmdkeys/erratumnamedlink/.setStrKeyWithDefFmt={#1}{}{%
    {{\@errata@getvalue{cmdkeys/uselabel}}+{\@errata@getvalue{cmdkeys/erratumname}}}},
%%  setCmdKeyArgsTwo={erratumnamedlink}{#1+#2}{{labelref}+{erratumname}},
  cmdkeys/erratumlistitemtitle/.setStrKeyWithDefFmt={#1}{defvalue}{%
    {{\@errata@getvalue{labelref}}+{\@errata@labelname}}},
  cmdkeys/erratumlisttarget/.setStrKeyWithDefFmt={#1}{}{{%
      \@errata@htargetlisttname{\the@errata@erratum}}},%
%%
  handleerratanamecmdkey/.code={%
    \StrKeyFmtgetformatted{errformats/errfmt}{\arabic{@errata@erratum}}{\@errata@erratumname}%
    \@errata@setvalues[cmdkeys/]{erratumname/.expand once={\@errata@erratumname}}%
  },
  handleactioncmdkeys/.code={%
    \@errata@setvalues{cmdkeys/actionid/.get=\@errata@tmp@actionid}%
    \@errata@ifelseempty{\@errata@tmp@actionid}{%
      \@errata@setvalues[cmdkeys/]{actiondescription={}}%
      \@errata@setvalues[cmdkeys/]{action={}}%
    }{%
      \@errata@setvalues{cmdkeys/actionargs/.get=\@errata@tmp@actionargs}%
      \StrKeyFmtgetformattedeo{actionformat/\@errata@tmp@actionid}{\@errata@tmp@actionargs}{\@errata@tmpf
      \@errata@setvalues[cmdkeys/]{actiondescription/.expand once={\@errata@tmpfmtvalue}}%
      \StrKeyFmtgetformattedeo{actionidformat/\@errata@tmp@actionid}{}{\@errata@tmpfmtvalue}%
      \@errata@setvalues[cmdkeys/]{action/.expand once={\@errata@tmpfmtvalue}}%
    }%
  },
  handlelabelcmdkey/.code={%
    \@errata@setvalue{prevlabel=\@errata@labelname}%
    \@errata@setlabelref{}%
    \ifboolexpr{ test {\ifdorecord} }{%
      %% Compare current label with previous. Print only if not already printed
      \ifnumcomp{\value{@errata@uniquelistlabels}}{>}{0}%
      {% True
```

```
        \ifthenelse{\expandafter\equal{\@errata@lastuniquelabel}{\@errata@labelname}}%
        {\@errata@setvalue{labelname={}}% Hide the label, if it equals the previous label
          \@errata@setvalue{newlabelgroup={false}}}%
        {\@errata@setvalue{prevlabel/.expand once=\@errata@labelname}%
          \@errata@setvalue{lastuniquelabel/.expand once=\@errata@labelname}%
          \@errata@setvalue{newlabelgroup={true}}%
        }%
      }{\@errata@setvalue{prevlabel=\@errata@labelname}}%
    }{}%
  },
  handlecustomformatkeys/.code={%
    \@errata@ifnotempty{\@errata@getvalue{customformatkeys}}{%
      \pgfkeysalso{setcmdkeyvalues/.list={\@errata@getvalue{customformatkeys}}}%
    }%
  },
  commandarghandlers/.code={%
    \pgfkeysalso{%
      handleerratanamecmdkey,
      handlelabelcmdkey,
      setcmdkeyvalues/.list={pageref, lineref},
      setcmdkeyvalues/.list={erratumnamedlink, erratumlistitemtitle, erratumlisttarget},
      handleactioncmdkeys,
      handlecustomformatkeys,
    }%
  },
  addtoargstring/.code={%
    \protected@xappto\@errata@argsstring{, #1={\@errata@getvalue{cmdkeys/#1}}}%
  },
  addcustomkeystoargsstring/.code={%
    \@errata@ifnotempty{\@errata@getvalue{customformatkeys}}{%
      \pgfkeysalso{addtoargstring/.list={{\@errata@getvalue{customformatkeys}}}}%
    }%
  },
  addtoargsstring/.style={%
    addtoargstring/.list={actiondescription, action, pageref, lineref,
      erratumlistitemtitle, erratumnamedlink, erratumlisttarget},
    addcustomkeystoargsstring,
  },
}
%%
\NewDocumentCommand{\@erratum@cmdinit}{}{%
  \@erratum@setprevispace% Handle if there was a previous white space glue
  \@erratum@checkenvironment%
}
%% ARG1: cmddefaultargs string
%% ARG2: Key/Args from cmd
%% ARG3: extra Key/Args
\NewDocumentCommand{\@erratum@cmd}{ m m m } {%
\refstepcounter{@errata@erratum}%
\@errata@setcmdkeys[]{\@errata@formatkeys}% Reset all values
```

```
%% Apply the default args
\@errata@setvalues[cmdkeys/]{defaultargs, #1, uselabel={\@errata@htargetname{\the@errata@erratum}}}%
\@errata@setvalues[cmdkeys/]{#2, #3}%
%% Execute the command key handlers
\erratataset{commandarghandlers}%
%% Define macro for the args string
\protected@edef\@errata@argsstring{#2}%
\@errata@ifnotempty{\@errata@argsstring}{%
  \eappto\@errata@argsstring{, }%
}%
\protected@eappto\@errata@argsstring{#3}%
%% Add rest of the arguments to \@errata@argsstring
\erratataset{addtoargsstring}%
}
%%
%% Macros for handling spaces before and after macro call
%%
\newcommand\ifstrequalexpand{\expandafter\ifstrequal\expandafter}%
%%
%% Handle if space after macro call should be ignored
\NewDocumentCommand{\@erratum@handleprevisspace}{}{%
\ifhmode%
\ifboolexpr{ bool {@errata@prevspace} and test {\ifstrequalexpand{\lastcommandinlinevalue}{}} }{%
\ignorespaces}{}%
\fi%
}
%%
\newif\if@errata@prevspace
%%
\NewDocumentCommand{\@erratum@setprevispace}{}{%
\ifhmode%
\ifdim \lastskip=\z@%
\@errata@prevspacefalse%
\else%
\@errata@prevspacetrue%
\fi%
\fi%
}
```

User commands for registering errata changes

`\erratumAdd` Macro for creating an Addition

```
\NewDocumentCommand{\@erratum@Add}{ O{} m }{% keyvals, new
\@erratum@cmdinit%
{% Group is necessary here to handle cref reference format for label type '@errata@erratum'.
%% \@erratum@cmd must be within the group, as pgfkeys are group local
\@erratum@cmd{defaultargsadd}{#1}{actionid={add}, actionargs={#2}}%
\@errata@dolayoutonpage{a}{}{#2}{\@errata@FormatKeywords{footformats}{\@errata@argsstring}{\@errata@footf
\@errata@dorecord{listformats}{\@errata@listformat}{\@errata@argsstring}%
}%
%% Handle if the space after the macro call should be ignored
```

```
\@erratum@handleprevisspace%
}
```

\erratumReplace Macro for creating a Replacement

```
\NewDocumentCommand{\@erratum@Replace}{ O{} m m }{% keyvals, old, new
\@erratum@cmdinit%
{% Group is necessary here to handle cref reference format for label type ‘@errata@erratum’.
%% \@erratum@cmd must be within the group, as pgfkeys are group local
\@erratum@cmd{defaultargsreplace}{#1}{actionid={replace}, actionargs={#3+#2}}%
\@errata@dolayoutonpage{r}{#2}{#3}{\@errata@FormatKeywords{footformats}{\@errata@argsstring}{\@errata@foo
\@errata@dorecord{listformats}{\@errata@listformat}{\@errata@argsstring}%
}%
%% Handle if the space after the macro call should be ignored
\@erratum@handleprevisspace%
}
```

\erratumDelete Macro for creating a Deletion

```
\NewDocumentCommand{\@erratum@Delete}{ O{} m }{% keyvals, removed
\@erratum@cmdinit%
{% Group is necessary here to handle cref reference format for label type ‘@errata@erratum’.
%% \@erratum@cmd must be within the group, as pgfkeys are group local
\@erratum@cmd{defaultargsdelete}{#1}{actionid={delete}, actionargs={#2}}%
\@errata@dolayoutonpage{d}{#2}{}{\@errata@FormatKeywords{footformats}{\@errata@argsstring}{\@errata@footf
\@errata@dorecord{listformats}{\@errata@listformat}{\@errata@argsstring}%
}%
\@erratum@handleprevisspace%
}
```

\erratumNote Macro for creating a Note entry

```
\NewDocumentCommand{\@erratum@Note}{ O{} m }{% keyvals, message
\@erratum@cmdinit%
{% Group is necessary here to handle cref reference format for label type ‘@errata@erratum’.
%% \@erratum@cmd must be within the group, as pgfkeys are group local
\@erratum@cmd{defaultargsnote}{#1}{actionid={note}, actionargs={#2}}%
\@errata@dolayoutonpage{n}{}{#2}{\@errata@FormatKeywords{footformats}{\@errata@argsstring}{\@errata@footf
\@errata@dorecord{listformats}{\@errata@listformat}{\@errata@argsstring}%
}%
%% Handle if the space after the macro call should be ignored
\@erratum@handleprevisspace%
}

%% Define with mandatory description argument if the requirecmddescription option is true
\if@erratumOpt@requirecmddescription
\NewDocumentCommand{\erratumAdd}{ O{} m m }{% keyvals, description, new
  \@erratum@Add[#1,description={#2}]{#3}}%
\NewDocumentCommand{\erratumReplace}{ O{} m m m }{% keyvals, description, old, new
\@erratum@Replace[#1,description={#2}]{#3}{#4}}%
\NewDocumentCommand{\erratumDelete}{ O{} m m }{% keyvals, description, removed
  \@erratum@Delete[#1,description={#2}]{#3}}%
\NewDocumentCommand{\erratumNote}{ O{} m m }{% keyvals, description, note
```

```
  \@erratum@Note[#1,description={#2}]{#3}}%
%% And define without description if the requirecmddescription option is false
\else
\let\erratumDelete\@erratum@Delete
\let\erratumAdd\@erratum@Add
\let\erratumReplace\@erratum@Replace
\let\erratumNote\@erratum@Note
\fi
```

9 The Errata of this Document

3	**Err(1)**	Page 4 Addition (forgotten word) three (added)
	Err(2)	Page 4 Deletion (superfluous phrase) darned (deleted)
	Err(3)	Page 4 Correction (translated) line (original text was: “Zeile”)
	Err(4)	Page 5 (old should be new)2006-11-21
7	**Err(5)**	Page 11 Addition (label) custom label example (added)
	Err(5)	Page 11 Addition (label) custom label example (added)
	Err(6)	Page 12 Addition (added comma) , (added)
	Err(7)	Page 12 Addition (added another comma) , (added)
	Err(8)	Page 12 Addition (added missing word) these (added)
	Err(9)	Page 12 Addition (added another missing word) corrections (added)

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the definition; numbers in roman refer to the pages where the entry is used.

Symbols

\@errata@dorecord 25, 26, 31, 32
\@errata@record 26, 26
\@errata@write 26, 26

A

action key 6
actiondescription key . . 6

C

cleveref key 1
cleveref (pkgoption) 14

D

date key 6
defaultargs key 3
defaultargs (pkgoption) . 15
defaultargsadd key 3
defaultargsadd (pkgoption) 15
defaultargsdelete key . . 3
defaultargsdelete (pkgoption) 15
defaultargsnote key 3
defaultargsnote (pkgoption) 16
defaultargsreplace key . . 3
defaultargsreplace (pkgoption) 15
description key 6

E

environments:
errata 9, 25
erratum 4, 25
environmenttestmode (pkgoption) 15
errata (environment) . . 9, 25
\erratamessage 9
\erratataRegisterFormatKey 18
\erratataSetFootFormat . . . 7
\erratataSetKeyFormat 7, 18, 18
\erratataSetListFormat . . . 7
\erratatemessage 26
erratum (environment) . 4, 25
\erratumAdd . . . 3, 31, 32, 33
\erratumDelete 3, 32, 32, 33
\erratumItem 9, 24, 26
erratumlistitem key 6
erratumlistitemtitle key 6
erratumlisttarget key . . 7
\erratumNote . . 4, 32, 32, 33
\erratumReplace 4, 32, 32, 33

F

foot key 5
foots key 1
foots (pkgoption) 14

H

hide key 1
hide (pkgoption) 14
hyperref key 1
hyperref (pkgoption) 14

I

inlinechange key 5
inlinechanges key 1
inlinechanges (pkgoption) 15
inlinemark key 5
inlinemarks key 1
inlinemarks (pkgoption) . 15

K

Keys
action 6
actiondescription . . . 6
cleveref 1
date 6
defaultargs 3
defaultargsadd 3
defaultargsdelete . . . 3
defaultargsnote 3
defaultargsreplace . . 3
description 6
erratumlistitem 6
erratumlistitemtitle 6
erratumlisttarget . . . 7
foot 5
foots 1
hide 1
hyperref 1
inlinechange 5
inlinechanges 1
inlinemark 5
inlinemarks 1
label 5
linenos 2
location 6
margin 5
marginargs 5
marginfmt/marginnote 7
marginnote 2
margins 1
pageref 6
record 1
reported-by 6
requirecmddescription 2
show 1
styles 2
type 6
uselabel 5
usemarginnote 5

L

label key 5
linenos key 2
linenos (pkgoption) 15
location key 6

M

margin key 5
marginargs key 5
marginfmt/marginnote key 7
marginnote key 2
marginnote (pkgoption) . . 14
margins key 1
margins (pkgoption) 14

P

package options:
cleveref 14
defaultargs 15

defaultargsadd 15
defaultargsdelete . . . 15
defaultargsnote 16
defaultargsreplace . . 15
environmenttestmode . 15
foots 14
hide 14
hyperref 14
inlinechanges 15
inlinemarks 15
linenos 15
marginnote 14
margins 14
record 14
requirecmddescription 14
show 14
styles 15
uniquelistlabels 15
pageref key 6
\PrintErrata 8, 24
\printerrata 9, 24, 24

R

record key 1
record (pkgoption) 14
reported-by key 6
requirecmddescription key 2
requirecmddescription (pkgoption) 14

S

show key 1
show (pkgoption) 14
styles key 2
styles (pkgoption) 15

T

type key 6

U

uniquelistlabels (pkgoption) 15
uselabel key 5
usemarginnote key 5

Change History

v0.1
General: First Version with Documentation 1
v0.2
General: More Documentation and Keywords Management 1
v0.3
General: adding type Keyword 1
v0.4
General: moved to GIT, fixing switches . . 1
v1.0
General: Added support for links with hyperref, and customization through formatters 1
cleveref package option added 14
defaultargs package option added . . . 15
defaultargsadd package option added 15
defaultargsdelete package option added 15
defaultargsreplace package option added 15
environmenttestmode package option added 15
hyperref package option added 14
inlinechanges package option added . 15
inlinemarks package option added . . . 15
linenos package option added 14
marginnote package option added . . . 14
requirecmddescription package option added 14
styles package option added 15
uniquelistlabels package option added 15