The **color-edits** package*

David Kempe
David.M.Kempe@gmail.com

September 19, 2020

Abstract

The color-edits style file packages and streamlines/improves a functionality that I and many co-authors used to copy-paste and redo by hand: marking/annotating edits by different authors in a jointly edited document. It provides one command that generates a suite of six commands for each author, assigning each author a color. These six automatically generated commands allow each author to distinguish by color text which the author suggests (1) adding, (2) substituting for other text, (3) deleting, (4) deleting while giving a reason for the deletion. In addition, it provides commands to add comments in the text or margin.

The package has two useful options. One option `suppress` allows to suppress all colors, replacements, comments, and marked deletions — it basically results in the document being displayed as you probably want it to when all suggested changes have been resolved. The option `showdeletions` shows all suggested deletions in the text in gray, rather than just indicating that *something* was deleted.

1 Introduction

The package `color-edits` implements a suite of functionalities that is often needed when multiple authors jointly edit a document. When doing so, authors will frequently want to mark up their (suggested or implemented) changes to the document in a way that is directly visible to co-authors, and also sometimes provide explanations for their changes, or raise issues/questions for their co-authors to consider or address. Many author teams address this by defining a macro for each author that displays the author's edits in an author-specific color. While this works well for additions, it does not make it as easy to mark deletions, and requires manual changes when comments have not been addressed and a deadline looms.

For example, if one has just one colorful macro for an author, and the author uses it both for marking recent edits/additions and providing comments to co-authors, there is no simple way to delete the comments while keeping the added

---

*This document corresponds to **color-edits** v1.1, dated 2020/09/18.

text (but changing its color to the "normal" one.) The present package tries to address these issues with a fairly simple implementation that provides six macros whose behavior can be altered in the (hopefully) "straightforward" way with package options.

2   Usage

2.1   Package Options

The package `color-edits` has two options: `suppress` and `showdeletions`. Here, we discuss their purpose at a high level — the precise effects are discussed in Section 2.4.

`suppress`   The option `suppress` suppresses all authors (see Section 2.3): it changes the definitions of the commands in such a way that the file that is produced looks as much as possible as one would expect if all editing issues had been addressed/accepted by the authors.

`showdeletions`   The option `showdeletions` is useful when one wants to see in the compiled file not only *that* something was replaced/deleted, but also *what* the old text was. Specifically, for the replacement and deletion macros, instead of just marking that something was deleted/replaced, it displays the replaced/deleted text in gray. Author suppression overrides `showdeletions`, i.e., `showdeletions` has no effect when an author is suppressed.

2.2   The exported macro \addauthor

The package only exports one macro: `\addauthor` [⟨*auth-name*⟩] {⟨*auth*⟩} {⟨*auth-color*⟩}. Calling this macro in the preamble of your document automatically generates the following six macros, described in more detail below.

1. `\<auth>edit` {⟨*text*⟩}

2. `\<auth>replace` {⟨*old-text*⟩} {⟨*new-text*⟩}

3. `\<auth>comment` {⟨*comment*⟩}

4. `\<auth>margincomment` {⟨*comment*⟩}

5. `\<auth>delete` {⟨*deleted-text*⟩}

6. `\<auth>deletecomment` {⟨*comment*⟩} {⟨*deleted-text*⟩}

When the optional argument ⟨*auth-name*⟩ is not provided, the argument ⟨*auth*⟩ will be used as the author's name. The intended use is that ⟨*auth*⟩ is short enough that authors will not mind using it as part of a command name, but ⟨*auth-name*⟩ might be more descriptive, to display with comments or notes about deletions. For example, if Ada Lovelace and Charles Babbage were working on a paper together, they might write the following:

```
\usepackage{color-edits}
\addauthor[Ada]{al}{blue}
\addauthor{cb}{green}
```

All edits by Ada will be marked in blue, and those by Charles in green. Whenever there is an attributed comment/deletion, for Ada, it would be prefaced by `Ada`, whereas for Charles, it would be prefaced by `cb`.

There is one specific use of the optional name. If an author's optional name ⟨*auth-name*⟩ is set to `suppress`, then this author's edits/comments/deletions are suppressed; see Section 2.3. Notice that this gives more fine-grained control — for instance, it can let you focus on the most recent round of edits/comments, if a new "author" has been added for it.[1]

2.3   Suppressed Authors

One sometimes wants to suppress the colorful edits and make the file look as much as possible as what is expected after all comments have been addressed and all suggested edits/deletions have been approved. We call this *suppressing* an author[2]. Specifically, suppressing an author does the following (described in more detail in the context of the specific macros):

- Remove all color.

- Remove all comments (both in the text and margins).

- For all text that has been (suggested as) deleted or replaced, nothing is displayed.

There are two ways to suppress authors: one can suppress individual authors by changing their optional names to `suppress` (see Section 2.2). Alternatively, by using the package option `suppress` (see Section 2.1), all authors are suppressed.

2.4   Generated Macros

In this section, we describe in detail the functionality of the macros generated by `\addauthor`. We will refer to them with the generic name. Throughout, keeping the earlier example of Ada and Charles in mind, the reader is encouraged to substitute in his/her mind `al` or `cb` for `<auth>`. We will frequently reference suppressed authors — see Section 2.3 for a discussion.

---

[1]I would like to thank Krishnamurthy Iyer for this suggestion and a suggested implementation.

[2]We still highly recommend explicitly addressing all issues, then removing the macros to let your co-author(s) know that you saw their edits and approve of them, and that all comments have been addressed. The use of suppression is intended primarily for a quick check of, e.g., how long your document will be once all comments are removed, or to focus on the most recent round of suggested edits. Of course, every now and then, it is 3 minutes before the submission deadline, and you have no choice. Please note that there may be some very minor spacing issues when using suppression — see Section 4.

The macros are internally defined using `\long\def`, so all arguments below that are given as ⟨*text*⟩ or ⟨*new-text*⟩ or ⟨*comment*⟩ (and similar ones) are allowed to contain multiple paragraphs.

`\<auth>edit` The macro `\<auth>edit {`⟨*text*⟩`}` in its default behavior simply displays ⟨*text*⟩ in the color assigned to the author `<auth>`. If the author has been suppressed, it displays ⟨*text*⟩ in the standard style/color. The macro is not affected by the `showdeletions` package option. The main purpose of this macro is to mark text as changed/added by the given author. If the text was changed, the implicit assumption is that the co-author will not care much about what the text was prior to the change — otherwise, the macro `\<auth>replace` would be preferable.

`\<auth>replace` The macro `\<auth>replace {`⟨*old-text*⟩`} {`⟨*new-text*⟩`}` in its default behavior simply displays ⟨*new-text*⟩ in the color assigned to the author `<auth>`. The macro should be used instead of `\<auth>edit` if the co-authors might want to see what was in the file before the change. While the old text is not displayed in the output per default, it is of course still in the LaTeX source, so co-authors can easily see it. If the author was suppressed, then `\<auth>replace` simply displays ⟨*new-text*⟩ in the standard style/color. If the package option `showdeletions` was used (and the author was not suppressed), then `\<auth>replace` displays ⟨*old-text*⟩ in gray, followed by ⟨*new-text*⟩ in the color assigned to the author `<auth>`.

`\<auth>comment` The macro `\<auth>comment {`⟨*text*⟩`}` is used for leaving comments for one's co-authors, as opposed to text that should become part of the document itself. Its default behavior is to show `[<auth-name>: <text>]` in the color assigned to the author `<auth>`. Here, ⟨*auth-name*⟩ is the name assigned to the author — either the optional long name if given, or otherwise ⟨*auth*⟩ (see Section 2.2). If the author is suppressed, then nothing is displayed. Notice that this is the main difference to the `\<auth>edit` macro, which still displays its text. This is in line with the idea that a change to the document/text should be included in a version that is — say — submitted to a conference, while internal comments to co-authors should not be included. The macro is not affected by the `showdeletions` package option.

`\<auth>margincomment` The macro `\<auth>margincomment {`⟨*text*⟩`}` is used for leaving comments for one's co-authors in the margins as opposed to in the main text. Thus, its purpose is basically the same as that of `\<auth>comment`, but different authors may prefer comments to be displayed in different places, e.g., for reasons of flow. Its default behavior is to show a marker in the main text in the color assigned to the author `<auth>`, and to show `[<auth-name>: <text>]` in the margin, also in the color assigned to the author `<auth>`. Here, ⟨*auth-name*⟩ is the name assigned to the author — either the optional long name if given, or otherwise ⟨*auth*⟩ (see Section 2.2). If the author is suppressed, then nothing is displayed. The macro is not affected by the `showdeletions` package option. Because the macro is implemented using the `\marginpar` macro, it cannot be used in environments where `\marginpar` does not work, such as in footnotes, section headings, or captions.

`\<auth>delete` The macro `\<auth>delete {`⟨*text*⟩`}` is used for letting co-authors know that one deleted ⟨*text*⟩ (or suggests doing so). Its default behavior is to display a marker in the color assigned to the author `<auth>` in the spot where the command occurs, and to leave a mark `[<auth-name> deleted here]` in the margin. Here, ⟨*auth-name*⟩ is the name assigned to the author — either the optional long name if given, or

4

otherwise ⟨*auth*⟩ (see Section 2.2). If the author is suppressed, then nothing is displayed. The main purpose of having the argument ⟨*text*⟩ is to allow co-authors to see not only *that* text was deleted, but *what* text was deleted. Accordingly, if the package option `showdeletions` is used (and the author is not suppressed), then instead of placing a marker and a comment in the margin, the text ⟨*text*⟩ is displayed, but in gray. This allows co-authors to see the deleted text not only in the source file, but also in the compiled output. Because the macro is implemented using the `\marginpar` macro, it cannot be used in environments where `\marginpar` does not work, such as in footnotes, section headings, or captions.

`\<auth>deletecomment`   The macro `\<auth>deletecomment {`⟨*comment*⟩`} {`⟨*text*⟩`}` is used for letting co-authors know that one deleted ⟨*text*⟩ (or suggests doing so), and to also provide a ⟨*comment*⟩ explaining the deletion. Its default behavior is to display a marker in the color assigned to the author `<auth>` in the spot where the command occurs, and to leave a mark `[<auth-name> deleted here: <comment>]` in the margin. Here, ⟨*auth-name*⟩ is the name assigned to the author — either the optional long name if given, or otherwise ⟨*auth*⟩ (see Section 2.2). If the author is suppressed, then nothing is displayed. As with the `\<auth>delete` command, the main purpose of having the argument ⟨*text*⟩ is to allow co-authors to see what text was deleted. If the package option `showdeletions` is used (and the author is not suppressed), then instead of placing a marker, the text ⟨*text*⟩ is displayed, but in gray. This allows co-authors to see the deleted text not only in the source file, but also in the compiled output. In addition, `[<auth-name>: <comment>]` is still displayed in the margin in the color assigned to the author. Because the macro is implemented using the `\marginpar` macro, it cannot be used in environments where `\marginpar` does not work, such as in footnotes, section headings, or captions.

3   The Package Implementation

3.1   The Initial Stuff

The package only requires the `ifthen` package (heavily, for distinctions based on package options and such) and the `color` package (obviously, to display stuff in colors for different authors).

```
1 \RequirePackage{ifthen}
2 \RequirePackage{color}
```

We define a color gray to use for showing deleted text. For all other colors, it is the user's responsibility to define them if they are not defined standard.

```
3 \definecolor{@gray}{rgb}{0.5,0.5,0.5}
```

3.2   Helper Macros for Defining the Exported Functions

`\coloredits@addauthoredit`   `\coloredits@addauthoredit {`⟨*auth*⟩`} {`⟨*auth-color*⟩`}` is a helper macro. It defines the command `\<auth>edit {`⟨*text*⟩`}`. How `\<auth>edit {`⟨*text*⟩`}` is defined depends on the package options and arguments. If the author is suppressed, then

`\<auth>edit {⟨text⟩}` simply displays ⟨text⟩. Otherwise, it displays ⟨text⟩ in the given color ⟨auth-color⟩.

```
4 \newcommand{\coloredits@addauthoredit}[3]{%
5 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
6 \expandafter\long\expandafter\def\csname #1edit\endcsname ##1{##1}%
7 }{% not suppressed
8 \expandafter\long\expandafter\def\csname #1edit\endcsname ##1{{\color{#3}##1}}
9 }}
```

`\coloredits@addauthorreplace`   `\coloredits@addauthorreplace {⟨auth⟩} {⟨auth-name⟩} {⟨auth-color⟩}` is a helper macro. It defines the command `\<auth>replace {⟨old-text⟩} {⟨new-text⟩}`. How `\<auth>replace {⟨old-text⟩} {⟨old-text⟩}` is defined depends on the package options and parameters passed in. If the author is suppressed, then `\<auth>replace {⟨old-text⟩} {⟨new-text⟩}` simply displays ⟨new-text⟩. If the author is not suppressed and showdeletions was not specified as a package option, it displays ⟨new-text⟩ in the given color ⟨auth-color⟩. If the author is not suppressed and showdeletions was specified as a package option, it displays ⟨old-text⟩ in gray, followed by ⟨new-text⟩ in the given color ⟨auth-color⟩.

```
10 \newcommand{\coloredits@addauthorreplace}[3]{%
11 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
12 \expandafter\long\expandafter\def\csname #1replace\endcsname ##1##2{##2}
13 }{% not suppressed
14 \ifthenelse{\equal{\coloredits@ShowDeletions}{yes}}{% Showing deletions
15 \expandafter\long\expandafter\def\csname #1replace\endcsname ##1##2{%
16 {\color{@gray}##1}{\color{#3}##2}}
17 }{% Not showing deletions
18 \expandafter\long\expandafter\def\csname #1replace\endcsname ##1##2{%
19 {\color{#3}##2}}
20 }%
21 }}
```

`\coloredits@addauthorcomment`   `\coloredits@addauthorcomment {⟨auth⟩} {⟨auth-name⟩} {⟨auth-color⟩}` is a helper macro. It defines the command `\<auth>comment {⟨text⟩}`. How `\<auth>comment {⟨text⟩}` is defined depends on the package options and arguments. If the author is suppressed, then `\<auth>comment {⟨text⟩}` does nothing. If the author is not suppressed, it displays, in square brackets, ⟨text⟩ in the given color ⟨auth-color⟩, prefixed by ⟨auth-name⟩.

```
22 \newcommand{\coloredits@addauthorcomment}[3]{%
23 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
24 \expandafter\long\expandafter\def\csname #1comment\endcsname ##1{}
25 }{% not suppressed
26 \expandafter\long\expandafter\def\csname #1comment\endcsname ##1{%
27 {\color{#3}[#2: ##1]}}
28 }}
```

`loredits@addauthormargincomment`   `\coloredits@addauthormargincomment {⟨auth⟩} {⟨auth-name⟩} {⟨auth-color⟩}` is a helper macro. It defines the command `\<auth>margincomment {⟨text⟩}`. How `\<auth>margincomment {⟨text⟩}` is defined depends on the package options and

arguments. If the author is suppressed, then \<auth>margincomment {⟨*text*⟩} does nothing. If the author is not suppressed, it displays, in the margin and in square brackets, ⟨*text*⟩ in the given color ⟨*auth-color*⟩, prefixed by ⟨*auth-name*⟩. It also places a bullet in square brackets of color ⟨*auth-color*⟩ in the text at the place where the command was used.

```
29 \newcommand{\coloredits@addauthormargincomment}[3]{%
30 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
31 \expandafter\long\expandafter\def\csname #1margincomment\endcsname ##1{}
32 }{% not suppressed
33 \expandafter\long\expandafter\def\csname #1margincomment\endcsname##1{{%
34 \color{#3}$[\bullet]$\marginpar{\scriptsize\color{#3}[#2: ##1]}}}
35 }}
```

\coloredits@addauthordelete   \coloredits@addauthordelete {⟨*auth*⟩} {⟨*auth-name*⟩} {⟨*auth-color*⟩} is a helper macro. It defines the command \<auth>delete {⟨*text*⟩}. How \<auth>delete {⟨*text*⟩} is defined depends on the package options and arguments. If the author is suppressed, then \<auth>delete {⟨*text*⟩} does nothing. If the author is not suppressed and showdeletions was not used, it places a bullet in brackets in the color ⟨*auth-color*⟩ at the place of the command, and places the note [<auth-name> deleted here] in the margin in the color ⟨*auth-color*⟩. If the author is not suppressed and showdeletions was used, it displays ⟨*text*⟩ in gray.

```
36 \newcommand{\coloredits@addauthordelete}[3]{%
37 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
38 \expandafter\long\expandafter\def\csname #1delete\endcsname ##1{}
39 }{% not suppressed
40 \ifthenelse{\equal{\coloredits@ShowDeletions}{yes}}{% Showing deletions
41 \expandafter\long\expandafter\def\csname #1delete\endcsname ##1{{\color{@gray}##1}}
42 }{% Not showing deletions
43 \expandafter\long\expandafter\def\csname #1delete\endcsname ##1{{%
44 \color{#3}$[\bullet]$\marginpar{\scriptsize\color{#3}#2 deleted here}}}
45 }%
46 }}
```

loredits@addauthordeletecomment   \coloredits@addauthordeletecomment {⟨*auth*⟩} {⟨*auth-name*⟩} {⟨*auth-color*⟩} is a helper macro. It defines the command \<auth>deletecomment {⟨*comment*⟩} {⟨*text*⟩}. How \<auth>deletecomment {⟨*comment*⟩} {⟨*text*⟩} is defined depends on the package options and arguments. If the author is suppressed, then \<auth>deletecomment {⟨*comment*⟩} {⟨*text*⟩} does nothing. If the author is not suppressed and showdeletions is not used, it places a bullet in brackets in the color ⟨*auth-color*⟩ at the place of the command, and places the note [<auth-name> deleted here: <comment>] in the margin in the color ⟨*auth-color*⟩. If the author is not suppressed and showdeletions was used, it displays ⟨*text*⟩ in gray, and places the note [<auth-name>: <comment>] in the margin in the color ⟨*auth-color*⟩.

```
47 \newcommand{\coloredits@addauthordeletecomment}[3]{%
48 \ifthenelse{\equal{\coloredits@SuppressEdits}{yes} \or \equal{#2}{suppress}}{% suppressed
49 \expandafter\long\expandafter\def\csname #1deletecomment\endcsname ##1##2{}
```

```
50 }{% not suppressed
51 \ifthenelse{\equal{\coloredits@ShowDeletions}{yes}}{% Showing deletions
52 \expandafter\long\expandafter\def\csname #1deletecomment\endcsname ##1##2{{%
53 \color{#3}$[\bullet]$\color{@gray}##2\marginpar{\scriptsize\color{#3}#2: ##1}}}
54 }{% Not showing deletions
55 \expandafter\long\expandafter\def\csname #1deletecomment\endcsname ##1##2{{%
56 \color{#3}$[\bullet]$\marginpar{\scriptsize\color{#3}#2 deleted here: ##1}}}
57 }%
58 }}
```

3.3 The Exported Macro \addauthor

\addauthor \addauthor [⟨*auth-name*⟩] {⟨*auth*⟩} {⟨*auth-color*⟩} is the only exported macro of the package. The definition of the \<auth>edit and \<auth>replace macros are independent of whether the optional longer name ⟨*auth-name*⟩ was provided, except for the author being suppressed if the longer name was suppress. For the definition of the two comment and deletion macros, there is a case distinction regarding whether the optional ⟨*auth-name*⟩ was not provided (in which case the short ⟨*auth*⟩ is used), or it was provided (in which case ⟨*auth-name*⟩ is used). For all the definitions, this macro just calls the helper macros to define the commands for the given author, using the given name and color.

```
59 \newcommand{\addauthor}[3][]{%
60 \coloredits@addauthoredit{#2}{#1}{#3}
61 \coloredits@addauthorreplace{#2}{#1}{#3}
62 \ifthenelse{\equal{#1}{}}{%optional name was not provided
63 \coloredits@addauthorcomment{#2}{#2}{#3}
64 \coloredits@addauthormargincomment{#2}{#2}{#3}
65 \coloredits@addauthordelete{#2}{#2}{#3}
66 \coloredits@addauthordeletecomment{#2}{#2}{#3}}{%optional name was provided
67 \coloredits@addauthorcomment{#2}{#1}{#3}
68 \coloredits@addauthormargincomment{#2}{#1}{#3}
69 \coloredits@addauthordelete{#2}{#1}{#3}
70 \coloredits@addauthordeletecomment{#2}{#1}{#3}}
71 }
```

Change History

4   Disclaimer and Known Problems

The biggest disclaimer is that I am not a very expert T_EXprogrammer, so this package may not be particularly well written. I am sharing it because several co-authors seem to enjoy using it and have encouraged me to share it publicly. If you see ways to improve the implementation or functionality (while keeping it still light-weight), I would love to hear from you. You are obviously using this package at your own risk.

As discussed above, because the macros for deletion and the one for comments in the margin are based on the `\marginpar` command in LaTeX, their use is restricted to settings in which `\marginpar` works. In particular, this excludes using them in footnotes, captions, and section/subsection titles. An alternative might be to use the `marginnote` package and its command `\marginnote`, but that package seems to have other known problems, and at the moment, I am not ready to try to decide which issues are larger. If you see a clean way to resolve this, feel free to let me know.

Earlier versions had some very minor spacing issues when using the `suppress` package option. Specifically, when there was a space before and after — say — an `\<auth>comment` command, then LaTeXwould still keep both spaces, resulting in slightly more space than there should be between the preceding and following items on the line. I may have accidentally fixed this in revisions, or the issue may still persist — I have not checked this extensively. Fixing this is not a high priority for me, as I view the suppression mechanism as primarily a way to see what the document will look like once all comments have been addressed/removed. An actual final version should actually address and remove them, so the spacing issues should be resolved in that way. If you notice the issue persisting and know of an easy way to resolve it, I'd love to hear from you and implement it.