

# **Das *changes*-Paket**

## **Manuelle Änderungsmarkierung – Version 3.0.0**

4. November 2018

**Ekkart Kleinod**

✉ [ekleinod@edgesoft.de](mailto:ekleinod@edgesoft.de)

## Inhalt

<b>1</b>	<b>Einleitung</b>	<b>3</b>
<b>2</b>	<b>Benutzung des <i>changes</i>-Pakets</b>	<b>4</b>
2.1	Verfügbare Skripte . . . . .	6
<b>3</b>	<b>Einschränkungen und Erweiterungsmöglichkeiten</b>	<b>8</b>
<b>4</b>	<b>Die Benutzerschnittstelle des <i>changes</i>-Pakets</b>	<b>9</b>
4.1	Paketooptionen . . . . .	9
4.1.1	draft . . . . .	9
4.1.2	final . . . . .	9
4.1.3	markup . . . . .	10
4.1.4	addedmarkup, deletedmarkup . . . . .	10
4.1.5	highlightmarkup . . . . .	11
4.1.6	commentmarkup . . . . .	12
4.1.7	authormarkup . . . . .	12
4.1.8	authormarkupposition . . . . .	13
4.1.9	authormarkuptext . . . . .	14
4.1.10	todonotes . . . . .	14
4.1.11	truncate . . . . .	14
4.1.12	ulem . . . . .	14
4.1.13	xcolor . . . . .	15
4.2	Änderungsmanagement . . . . .	15
4.2.1	\added . . . . .	15
4.2.2	\deleted . . . . .	16
4.2.3	\replaced . . . . .	16
4.3	Hervorhebungen und Kommentare . . . . .	16
4.3.1	\highlight . . . . .	16
4.3.2	\comment . . . . .	17
4.4	Änderungsübersicht . . . . .	17
4.4.1	\listofchanges . . . . .	17
4.5	Autorenverwaltung . . . . .	18
4.5.1	\definechangesauthor . . . . .	18
4.6	Anpassung der Ausgabe . . . . .	19
4.6.1	\setaddedmarkup . . . . .	19
4.6.2	\setdeletedmarkup . . . . .	19
4.6.3	\sethighlightmarkup . . . . .	20
4.6.4	\setcommentmarkup . . . . .	20
4.6.5	\setauthormarkup . . . . .	20
4.6.6	\setauthormarkupposition . . . . .	21
4.6.7	\setauthormarkuptext . . . . .	21
4.6.8	\settruncatewidth . . . . .	21

4.6.9	\setsummarywidth . . . . .	22
4.6.10	\setsummarytewidth . . . . .	22
4.6.11	\setsoceextension . . . . .	22
4.7	Benötigte Pakete . . . . .	23
<b>5</b>	<b>Autorinnen und Autoren</b>	<b>24</b>
<b>6</b>	<b>Versionen</b>	<b>25</b>
<b>7</b>	<b>Weitergabe, Copyright, Lizenz</b>	<b>26</b>

## 1 Einleitung

Dieses Paket dient dazu, manuelle Änderungsmarkierung zu ermöglichen.

Verbesserungsvorschläge, Gedanken oder Kritik sind willkommen. Das Paket wird auf *gitlab* gehalten, bitte gehen Sie zu

<http://edgesoft.de/projects/latex/changes/>

für Links zum Quellcodezugang, Fehler- und Featuretracker etc. Wenn Sie mich direkt kontaktieren wollen, mailen Sie bitte an [ekleinod@edgesoft.de](mailto:ekleinod@edgesoft.de). Bitte starten Sie Ihr Mail-Subject mit [changes].

README: Das changes-Paket dient zur manuellen Markierung von geändertem Text, insbesondere Einfügungen, Löschungen und Ersetzungen. Der geänderte Text wird farbig markiert und, bei gelöschtem Text, durchgestrichen. Zusätzlich kann Text hervorgehoben und/oder kommentiert werden. Das Paket ermöglicht die freie Definition von Autoren und deren zugeordneten Farben. Es erlaubt zusätzlich die Änderung des Änderungs-, Autor-, Hervorhebungs- und Kommentarmarkups.

Ein kurzes Beispiel für Änderungsmarkierung:

Das ist **zugefügter**<sup>EK</sup> Text. In diesem Satz ersetze ich ein **gutes**~~**schlechtes**~~<sup>[EK 1]: Weil ein gutes Wort besser ist</sup> Wort. Und jetzt noch ein **schlechtes**<sup>EK</sup> Wort zum Löschen. Text kann auch **hervorgehoben**<sup>EK</sup> oder nur **[EK 2]: Aus Spaß!** kommentiert werden.

## 2 Benutzung des *changes*-Pakets

In diesem Kapitel wird die Nutzung des *changes*-Pakets beschrieben. Dabei wird ein typischer Anwendungsfall geschildert. Die ausführliche Beschreibung der Paketoptionen und neuen Befehle finden Sie nicht hier, sondern in Kapitel 4.

Ausgangslage ist ein Text, an dem Änderungen vorgenommen werden sollen. Diese Änderungen sollen markiert werden, und zwar für jeden Autor einzeln. Eine solche Änderungsmarkierung ist z. B. von WYSIWYG-Textprogrammen wie *LibreOffice*, *OpenOffice* oder *Word* bekannt.

Zu diesem Zweck wurde das *changes*-Paket entwickelt. Das Paket stellt Befehle zur Verfügung, um verschiedene Autoren zu definieren und Text als zugefügt, gelöscht oder geändert zu markieren. Zusätzlich kann Text hervorgehoben oder kommentiert werden. Um das Paket zu nutzen, müssen Sie folgende Schritte ausführen:

1. *changes*-Paket einbinden
2. Autoren definieren
3. Textänderungen markieren
4. Text hervorheben und kommentieren
5. Dokument mit  $\LaTeX$  setzen
6. Liste von Änderungen anzeigen lassen
7. Markierungen entfernen

### *changes*-Paket einbinden

Um die Änderungsverfolgung zu aktivieren, ist das *changes*-Paket wie folgt einzubinden:

```
\usepackage{changes}
```

bzw.

```
\usepackage[<options>]{changes}
```

Mit den verfügbaren Optionen bestimmen Sie hauptsächlich das Aussehen der Änderungsmarkierungen. Sie können das Aussehen der Änderungsmarkierungen auch nach Einbinden des *changes*-Pakets verändern.

Für Details lesen Sie bitte Abschnitt 4.1 und Abschnitt 4.6.

### Autoren definieren

Das *changes*-Paket stellt einen vordefinierten anonymen Autor zur Verfügung. Wenn Sie jedoch die Änderungen per Autor\_in verfolgen wollen, müssen Sie die entsprechenden Autor\_innen definieren. Dies geht wie folgt:

```
\definechangesauthor[<options>]{<id>}
```

Über die ID werden der/die Autor\_in und die zugehörigen Textänderungen eindeutig identifiziert. Optional können Sie einen Namen angeben und dem/der Autor\_in eine eigene Farbe zuweisen.

Für Details lesen Sie bitte Abschnitt 4.5.

### Textänderungen markieren

Jetzt ist alles vorbereitet, um den geänderten Text zu markieren. Benutzen Sie bitte je nach Änderung die folgenden Befehle:

für neu zugefügten Text:

```
\added[id=<ID>, comment=<Kommentar>]{neuer Text}
```

für gelöschten Text:

```
\deleted[id=<ID>, comment=<Kommentar>]{alter Text}
```

für geänderten Text:

```
\replaced[id=<ID>, comment=<Kommentar>]{neuer Text}{alter Text}
```

Die Angabe von Autoren-ID und eines Kommentars ist optional.

Für Details lesen Sie bitte Abschnitt 4.2.

### Text hervorheben und kommentieren

Vielleicht möchten Sie noch Text hervorheben oder kommentieren?

Text hervorheben:

```
\highlight[id=<id>, comment=<comment>]{text}
```

Text kommentieren:

```
\comment[id=<ID>]{Kommentar}
```

Die Angabe der Autoren-ID und des Kommentars für Hervorhebungen ist optional.

Für Details lesen Sie bitte Abschnitt 4.3.

### Dokument mit $\LaTeX$ setzen

Nachdem Sie die Änderungen im  $\TeX$ -Text markiert haben, können Sie sie im erzeugten Dokument sichtbar machen, indem Sie das Dokument ganz normal übersetzen. Durch die Übersetzung wird der geänderte Text so markiert, wie Sie das mittels der Optionen bzw. speziellen Befehle eingestellt haben.

### Liste von Änderungen anzeigen lassen

Sie können sich eine Liste der Änderungen ausgeben lassen. Dies erfolgt mit dem Kommando:

```
\listofchanges[style=<list|summary|compactsummary>, title=<Titel>,  
show=<Typ>]
```

Die Ausgabe ist gedacht als Analogon zur Liste von Tabellen oder Abbildungen.

Die Angabe des Stils ist optional, standardmäßig wird `style=list` gewählt. Um einen schnellen Überblick über Art und Anzahl der Änderungen abhängig von dem/der Autor\_in zu bekommen, verwenden Sie den Befehl mit der Option `style=summary` oder `style=compactsummary`. Zeigen Sie nur bestimmte Änderungstypen mit der `show`-Option.

Bei jedem  $\TeX$ -Lauf werden die Daten für diese Liste in eine Hilfsdatei geschrieben. Beim nächsten  $\TeX$ -Lauf werden dann diese Daten genutzt, um die Änderungsliste anzuzeigen. Daher sind nach jeder Änderung zwei  $\TeX$ -Läufe notwendig, um eine aktuelle Änderungsliste anzuzeigen.

Für Details lesen Sie bitte Abschnitt 4.4.

### Markierungen entfernen

Oft ist es der Fall, dass die Änderungen eines Dokuments angenommen oder abgelehnt werden und nach diesem Prozess die Änderungsmarkierungen entfernt werden sollen. Sie können die Ausgabe der Änderungsmarkierungen per Option beim Einbinden des *changes*-Pakets unterdrücken:

```
\usepackage[final]{changes}
```

## 2.1 Verfügbare Skripte

Für die Entfernung der Markierungen aus dem Quelltext steht ein Script von Silvano Chiaradonna zur Verfügung. Das Script liegt im Verzeichnis:

```
<texpath>/scripts/changes/
```

Das Script entfernt alle Markierungen. Sie können die zu entfernenden Markierungen selektieren bzw. selektieren, indem Sie den interaktiven Modus einschalten. Der interaktive Modus wird mit dem Skriptparameter `-i` eingeschaltet.

Das Skript ist noch nicht an die Kommentar- und Hervorhebungsbefehle angepasst.



### 3 Einschränkungen und Erweiterungsmöglichkeiten

Das *changes*-Paket ist sorgfältig programmiert und getestet worden. Dennoch kann es vorkommen, dass Fehler im Paket sind, dass die Benutzung problematisch ist oder dass eine Funktion fehlt, die Sie gerne hätten. In diesem Fall gehen Sie bitte zu

<http://changes.sourceforge.net/>

Dort können Sie Fehler melden, im Forum um Hilfe fragen oder Tips einstellen. Sie können dort den Quellcode ansehen und nach Ihren Wünschen ändern bzw. erweitern. Ich werde mich dann bemühen, Ihre Änderungen einzuarbeiten. Sie können als Co-Autor am Paket mitarbeiten, wenn Sie bei *sourceforge* angemeldet sind.

Sie können mir auch eine Mail schreiben an [ekleinod@edgesoft.de](mailto:ekleinod@edgesoft.de), in diesem Fall starten Sie bitte Ihr Mail-Subject mit [changes].

Die Änderungsmarkierung von Text funktioniert recht gut, es können auch ganze Absätze markiert werden. Die Markierung von mehreren Absätzen gleichzeitig, von Bildern und Tabellen ist eingeschränkt oder nicht möglich.

Sie können versuchen, solchen Text in eine eigene Datei auszulagern, und diese mit *input* einzubinden. Manchmal hilft das, oft ist es einen Versuch wert. Danke an Charly Arenz für diesen Tip.

Fußnoten oder Randnotizen werden in bestimmten Umgebungen, z. B. Tabellen oder der *tabbing*-Umgebung, nicht korrekt gesetzt. Vermeiden Sie das Markup, wenn Sie diese Umgebungen benutzen.

Das Paket bietet Raum für Erweiterungen, die ich jedoch nicht selbst programmieren werde (weil mir Zeit und oft auch die Fähigkeit fehlt). Ich liste hier einige Möglichkeiten auf, eine komplettere Liste finden Sie auf der *gitlab*-Seite:

- Auswahl der anzunehmenden/abzulehnenden Änderungen mit entsprechendem Löschen des Textes
- Markierung von mehreren Absätzen
- Markierung von Bildern und Tabellen
- automatische Markierung anhand von diff-Informationen (unter Berücksichtigung der Einschränkungen bzgl. Absätzen, Bildern, etc.)
- Übersetzung der sprachabhängigen Texte und der Nutzerdokumentation in andere Sprachen

## 4 Die Benutzerschnittstelle des *changes*-Pakets

In diesem Kapitel wird die Nutzerschnittstelle des *changes*-Pakets vorgestellt, d. h. alle Optionen und Kommandos. Jede Option bzw. jedes neue Kommando werden beschrieben. Wenn Sie die Optionen und Kommandos im Beispiel sehen wollen, sehen Sie bitte in das Beispielverzeichnis unter

`<texpath>/doc/latex/changes/examples/`

Die Beispieldateien sind mit der benutzten Option bzw. dem benutzten Kommando benannt.

Eine Vorbemerkung zum Setzen von ersetzttem Text: ersetzter Text wird immer wie folgt gesetzt: `<neuer Text><alter Text>`. Daher gibt es keine Möglichkeit, die Ausgabe ersetzten Texts direkt zu beeinflussen, sondern nur über die Änderung der Ausgabe neuen bzw. gelöschten Texts.

### 4.1 Paketoptionen

#### 4.1.1 draft

Die *draft*-Option bewirkt, dass alle Änderungen markiert werden. Die Änderungsliste kann durch `\listofchanges` ausgegeben werden. Diese Option ist automatisch voreingestellt.

Die Angabe von *draft* in `\documentclass` wird vom *changes*-Paket mitgenutzt. Die lokale Angabe von *final* überstimmt die Angabe von *draft* in `\documentclass`.

`\usepackage[draft]{changes} \cong \usepackage{changes}`

#### 4.1.2 final

Die *final*-Option bewirkt, dass alle Änderungsmarkierungen ausgeblendet werden und nur noch der korrekte Text ausgegeben wird. Die Änderungsliste wird ebenfalls unterdrückt.

Die Angabe von *final* in `\documentclass` wird vom *changes*-Paket mitgenutzt. Die lokale Angabe von *draft* überstimmt die Angabe von *final* in `\documentclass`.

`\usepackage[final]{changes}`

### 4.1.3 markup

Die `markup`-Option wählt ein vordefiniertes visuelles Markup für geänderten Text. Das `default`-Markup wird gewählt, wenn die Option nicht gesetzt wird. Das mit `markup` gewählte Markup kann mit den spezielleren Optionen `addedmarkup`, `deletedmarkup`, `commentmarkup` oder `highlightmarkup` geändert werden.

Die folgenden Werte sind erlaubt:

`default`

default für zugefügten, gelöschten und hervorgehobenen Text sowie Kommentare (default-Markup)

`underlined`

zugefügter Text wird unterstrichen, gewellt unterstrichen für Hervorhebungen, default für gelöschten Text sowie Kommentare

`bfit`

fetter zugefügter Text, schräger gelöschter Text, default für hervorgehobenen Text sowie Kommentare

`nocolor`

es werden keine Farben verwendet, zugefügter Text wird unterstrichen, gewellt unterstrichen für Hervorhebungen, default für gelöschten Text sowie Kommentare

*Aufruf*

```
\usepackage[markup=markup]{changes}
```

*Beispiele*

```
\usepackage[markup=default]{changes} ≅ \usepackage{changes}
```

```
\usepackage[markup=underlined]{changes}
```

```
\usepackage[markup=bfit]{changes}
```

```
\usepackage[markup=nocolor]{changes}
```

Wenn von farbigem zu nichtfarbigem Markup oder umgekehrt gewechselt wird und eine Hilfsdatei existiert werden einige Kompilierfehler angezeigt. Über diese ist hinwegzuspringen, beim zweiten Durchlauf sollten die Fehler verschwunden sein.

### 4.1.4 addedmarkup, deletedmarkup

Die `addedmarkup`-Option wählt ein vordefiniertes visuelles Markup für zugefügten Text. Die `deletedmarkup`-Option wählt analog ein vordefiniertes visuelles Markup für gelöschten Text. Das `default`-Markup wird gewählt, wenn die Option nicht gesetzt wird. Die Optionen `addedmarkup` und `deletedmarkup` überschreiben das mit `markup` gewählte Markup.

Die folgenden Werte sind erlaubt:

colored

kein Textmarkup, nur farbige Kennzeichnung – **Beispiel** (default-Markup für zugefügten Text)

uline

unterstrichener Text – Beispiel

uuline

doppelt unterstrichener Text – Beispiel

uwave

gewellt unterstrichener Text – Beispiel

dashuline

gestrichelt unterstrichener Text – Beispiel

dotuline

gepunktet unterstrichener Text – Beispiel

sout

durchgestrichener Text (nur gelöschter Text) – ~~Beispiel~~ (default-Markup für gelöschten Text)

xout

schräg durchgestrichener Text (nur gelöschter Text) – ~~Beispiel~~

bf fetter Text – **Beispiel**

it italic Text – *Beispiel*

sl schräger Text – *Beispiel*

em hervorgehobener Text – *Beispiel*

*Aufruf*

```
\usepackage[addedmarkup=<markup>]{changes}
```

*Beispiele*

```
\usepackage[addedmarkup=colored]{changes} \equiv \usepackage{changes}
```

```
\usepackage[addedmarkup=uline]{changes}
```

*Aufruf*

```
\usepackage[deletedmarkup=<markup>]{changes}
```

*Beispiele*

```
\usepackage[deletedmarkup=sout]{changes} \equiv \usepackage{changes}
```

```
\usepackage[deletedmarkup=xout]{changes}
```

```
\usepackage[deletedmarkup=uwave]{changes}
```

#### 4.1.5 highlightmarkup

Die `highlightmarkup`-Option wählt ein vordefiniertes visuelles Markup für hervorgehobenen Text. Das default-Markup wird gewählt, wenn die Option nicht gesetzt wird. Die Option `highlightmarkup` überschreibt das mit `markup` gewählte Markup.

Die folgenden Werte sind erlaubt:

background

Hervorhebung durch Hintergrundfarbe – Beispiel (default-Markup für Hervorhebungen)

uuline

doppelt unterstrichener Text – Beispiel

uwave

gewellt unterstrichener Text – Beispiel

*Aufruf*

```
\usepackage[highlightmarkup=<markup>]{changes}
```

*Beispiele*

```
\usepackage[highlightmarkup=background]{changes} ≡ \usepackage{changes}
```

```
\usepackage[highlightmarkup=uwave]{changes}
```

#### 4.1.6 commentmarkup

Die `commentmarkup`-Option wählt ein vordefiniertes visuelles Markup für Kommentare. Das default-Markup wird gewählt, wenn die Option nicht gesetzt wird. Die Option `commentmarkup` überschreibt das mit `markup` gewählte Markup.

Die folgenden Werte sind erlaubt:

todo

**Beispielkommentar** Kommentar als ToDo-Notiz, die nicht in der Liste der Todos erscheint – (default-Markup für Kommentare)

margin

Beispielkommentar Kommentar im Seitenrand –

footnote

Kommentar als Fußnote – <sup>1</sup>

uwave

gewellt unterstrichener Text – Beispielkommentar

*Aufruf*

```
\usepackage[commentmarkup=<markup>]{changes}
```

*Beispiele*

```
\usepackage[commentmarkup=todo]{changes} ≡ \usepackage{changes}
```

```
\usepackage[commentmarkup=footnote]{changes}
```

#### 4.1.7 authormarkup

Die `authormarkup`-Option wählt ein vordefiniertes visuelles Markup für die Autor-Identifizierung. Das default-Markup wird gewählt, wenn die Option nicht gesetzt wird.

---

<sup>1</sup> Beispielkommentar

Die folgenden Werte sind erlaubt:

**superscript**

hochgestellter Text – Text<sup>Beispiel</sup> (default-Markup)

**subscript**

tiefgestellter Text – Text<sub>Beispiel</sub>

**brackets**

Text in Klammern – Text(Beispiel)

**footnote**

Text in einer Fußnote – Text<sup>2</sup>

**none**

keine Autor-Identifizierung

*Aufruf*

```
\usepackage[authormarkup=<markup>]{changes}
```

*Beispiele*

```
\usepackage[authormarkup=superscript]{changes} ≡ \usepackage{changes}
```

```
\usepackage[authormarkup=subscript]{changes}
```

```
\usepackage[authormarkup=brackets]{changes}
```

```
\usepackage[authormarkup=footnote]{changes}
```

```
\usepackage[authormarkup=none]{changes}
```

#### 4.1.8 authormarkupposition

Die *authormarkupposition*-Option gibt an, wo die Autor-Identifizierung gesetzt wird. Der default-Wert wird gewählt, wenn die Option nicht gesetzt wird.

Die folgenden Werte sind erlaubt:

**right**

rechts vom Text – Text<sup>Beispiel</sup> (default value)

**left**

links vom Text – <sup>Beispiel</sup>Text

*Aufruf*

```
\usepackage[authormarkupposition=<markup>]{changes}
```

*Beispiele*

```
\usepackage[authormarkupposition=right]{changes} ≡ \usepackage{changes}
```

```
\usepackage[authormarkupposition=left]{changes}
```

---

<sup>2</sup> Beispiel

#### 4.1.9 authormarkuptext

Die *authormarkuptext*-Option gibt an, was für die Autor-Identifizierung genutzt wird. Der default-Wert wird gewählt, wenn die Option nicht gesetzt wird.

Die folgenden Werte sind erlaubt:

*id*   Autoren-ID – Text<sup>ID</sup> (default-Wert)

*name*       Autorenname – Text<sup>Autorenname</sup>

*Aufruf*

```
\usepackage[authormarkuptext=<markup>]{changes}
```

*Beispiele*

```
\usepackage[authormarkuptext=id]{changes} ≅ \usepackage{changes}
```

```
\usepackage[authormarkuptext=name]{changes}
```

#### 4.1.10 todonotes

Optionen für das *todonotes*-Paket können als Parameter der *todonotes*-Option angegeben werden. Zwei oder mehr Optionen müssen in geschweifte Klammern gesetzt werden.

*Aufruf*

```
\usepackage[todonotes=<options>]{changes}
```

*Beispiele*

```
\usepackage[todonotes={textsize=tiny}]{changes}
```

#### 4.1.11 truncate

Optionen für das *truncate*-Paket können als Parameter der *truncate*-Option angegeben werden. Zwei oder mehr Optionen müssen in geschweifte Klammern gesetzt werden.

*Aufruf*

```
\usepackage[truncate=<options>]{changes}
```

*Beispiele*

```
\usepackage[truncate=hyphenate]{changes}
```

#### 4.1.12 ulem

Optionen für das *ulem*-Paket können als Parameter der *ulem*-Option angegeben werden. Zwei oder mehr Optionen müssen in geschweifte Klammern gesetzt werden.

*Aufruf*

```
\usepackage[ulem=<options>]{changes}
```

*Beispiele*

```
\usepackage[ulem=normalem]{changes}
```

```
\usepackage[ulem={normalem,normalbf}]{changes}
```

### 4.1.13 xcolor

Optionen für das *xcolor*-Paket können als Parameter der *xcolor*-Option angegeben werden. Zwei oder mehr Optionen müssen in geschweifte Klammern gesetzt werden.

*Aufruf*

```
\usepackage[xcolor=<options>]{changes}
```

*Beispiele*

```
\usepackage[xcolor=dvipdf]{changes}
```

```
\usepackage[xcolor={dvipdf,gray}]{changes}
```

## 4.2 Änderungsmanagement

Alle Beispiele nutzen als Kommentarfomatierung *uwave*, da Randnotizen und ToDo-Notizen in *tabbing*-Umgebungen nicht erlaubt sind (siehe Kapitel 3).

### 4.2.1 \added

`\added`

Der Befehl `\added` markiert zugefügten Text. Der neue Text wird als notwendiges Argument in geschweiften Klammern übergeben.

Das optionale Argument enthält Key-Value-Paare für die Angabe von Autor-ID sowie eines Kommentars. Die Autor-ID muss mit einer mit dem `\definechangesauthor`-Befehl definierten ID übereinstimmen. Enthält der Kommentar Sonderzeichen oder Leerzeichen, ist er in geschweifte Klammern einzuschließen.

Wenn ein Kommentar angegeben wurde, wird das direkte Autormarkup am geänderten Text unterdrückt, da es im Kommentar erscheint.

*Aufruf*

```
\added[id=<Autor-ID>, comment=<Kommentar>]{<neuer Text>}
```

*Beispiele*

```
Das ist \added[id=EK]{neuer} Text.
```

```
Das ist neuerEK Text.
```

```
Das ist \added[id=EK, comment={muss rein}]{neuer} Text.
```

```
Das ist neuer[EK 3]: muss rein Text.
```

```
Das ist \added[comment=anonym]{neuer} Text.
```



Das ist neuer[1]: anonym Text.

### 4.2.2 `\deleted`

`\deleted`

Der Befehl `\deleted` markiert gelöschten Text. Argumente: siehe `\added`.

*Aufruf*

```
\deleted[id=<Autor-ID>, comment=<Kommentar>]{<gelöschter Text>}
```

*Beispiele*

Das ist `\deleted[comment=obsolet]{schlechter}` Text.

Das ist schlechter[2]: obsolet Text.

### 4.2.3 `\replaced`

`\replaced`

Der Befehl `\replaced` markiert geänderten Text. Notwendige Argumente sind der neue sowie der alte Text. Optionale Argumente: siehe `\added`.

*Aufruf*

```
\replaced[id=<Autor-ID>, comment=<Kommentar>]{<neuer Text>}{<alter Text>}
```

*Beispiele*

Das ist `\replaced[id=EK]{schöner}{schlechter}` Text.

Das ist schöner~~schlechter~~<sup>EK</sup> Text.

## 4.3 Hervorhebungen und Kommentare

### 4.3.1 `\highlight`

`\highlight`

Der Befehl `\highlight` markiert hervorgehobenen Text. Der hervorzuhebende Text wird als notwendiges Argument in geschweiften Klammern übergeben.

Das optionale Argument enthält Key-Value-Paare für die Angabe von Autor-ID sowie eines Kommentars. Die Autor-ID muss mit einer mit dem `\definechangesauthor`-Befehl definierten ID übereinstimmen. Enthält der Kommentar Sonderzeichen oder Leerzeichen, ist er in geschweifte Klammern einzuschließen.

Wenn ein Kommentar angegeben wurde, wird das direkte Autormarkup am geänderten Text unterdrückt, da es im Kommentar erscheint.

*Aufruf*

```
\highlight[id=<Autor-ID>, comment=<Kommentar>]{<hervorgehobener Text>}
```

*Beispiele*

Das ist \highlight[id=EK, comment={Mein Kommentar}]{hervorgehobener} Text.

Das ist hervorgehobener [\[EK 4\]: Mein Kommentar](#) Text.

Das ist \highlight[comment={Anonymer Kommentar}]{anonym hervorgehobener} Text.

Das ist anonym hervorgehobener [\[3\]: Anonymer Kommentar](#) Text.

### 4.3.2 \comment

`\comment`

Der Befehl `\comment` fügt dem Dokument einen Kommentar hinzu. Der Kommentar wird als notwendiges Argument in geschweiften Klammern übergeben.

Das optionale Argument enthält ein Key-Value-Paar für die Angabe der Autor-ID. Die Autor-ID muss mit einer mit dem `\definechangesauthor`-Befehl definierten ID übereinstimmen.

Die Kommentare werden durchnummeriert, die Nummer erscheint im Kommentar.

*Aufruf*

```
\comment[id=<Autor-ID>]{<Kommentar>}
```

*Beispiele*

Das ist kommentierter\comment[id=EK]{Mein Kommentar} Text.

Das ist kommentierter[\[EK 5\]: Mein Kommentar](#) Text.

Das ist kommentierter\comment{Anonymer Kommentar} Text.

Das ist kommentierter[\[4\]: Anonymer Kommentar](#) Text.

## 4.4 Änderungsübersicht

### 4.4.1 \listofchanges

`\listofchanges`

Der Befehl `\listofchanges` gibt eine Liste oder Zusammenfassung der Änderungen aus. Im ersten  $\TeX$ -Lauf wird eine Hilfsdatei angelegt, deren Daten im zweiten Durchlauf eingebunden werden. Für eine aktuelle Liste der Änderungen sind daher zwei  $\TeX$ -Läufe notwendig.

Es können drei optionale Argumente angegeben werden:

- style
- title
- show

**style** Über das Argument `style` können verschiedene Listenstile für die Anzeige ausgewählt werden. Es sind folgende drei Stile erlaubt:

- `list` – gibt die Änderungsliste wie ein Inhaltsverzeichnis aus (default)
- `summary` – gibt die Anzahl der Änderungen gruppiert nach Autor aus
- `compactsummary` – wie `summary`, jedoch werden Änderungen mit Anzahl 0 nicht ausgegeben

**title** Mit dem Argument `title` kann ein eigener Titel für die Änderungsliste angegeben werden. Wenn Sie Sonderzeichen oder Leerzeichen im Titel benutzen wollen, klammern Sie den Titel geschweift ein.

**show** Das Argument `show` gibt an, welche Änderungstypen in der Änderungsliste ausgegeben werden. Die folgenden Werte sind erlaubt:

- `all` – alle Typen (default)
- `added` – nur neue Texte
- `deleted` – nur Löschungen
- `replaced` – nur Ersetzungen
- `highlight` – nur Hervorhebungen
- `comment` – nur Kommentare

Sie können die Werte mit Hilfe des Zeichens `|` kombinieren. Wenn Sie z. B. alle neuen Texte und alle Löschungen anzeigen wollen, geben Sie `show=added|deleted` an.

*Aufruf*

```
\listofchanges[style=<list|summary|compactsummary>, title=<Titel>,  
show=<Typ>]
```

*Beispiele*

```
\listofchanges  
\listofchanges[style=list] ≡ \listofchanges  
\listofchanges[style=summary, title={Meine Zusammenfassung}]  
\listofchanges[title={Kommentare}, show=comment]
```

## 4.5 Autorenverwaltung

### 4.5.1 `\definechangesauthor`

`\definechangesauthor`

Der Befehl `\definechangesauthor` definiert einen neuen Autor/eine neue Autorin für Änderungen. Es muss eine eindeutige Autor-ID angegeben werden, die keine Sonder- oder Leerzeichen enthalten darf. Optional kann eine Farbe und ein Name angegeben werden.

Wird keine Farbe angegeben, wird blau genutzt. Der Autor\_innenname wird in der Änderungsliste sowie im Markup benutzt, im Markup jedoch nur, wenn die entsprechende Option gesetzt ist.

*Aufruf*

```
\definechangesauthor[name={\author's name}, color={\color}]{\author's id}
```

*Beispiele*

```
\definechangesauthor{EK}
\definechangesauthor[color=orange]{EK}
\definechangesauthor[name={Ekkart Kleinod}]{EK}
\definechangesauthor[name={Ekkart Kleinod}, color=orange]{EK}
```

## 4.6 Anpassung der Ausgabe

### 4.6.1 \setaddedmarkup

`\setaddedmarkup`

Der Befehl `\setaddedmarkup` legt fest, wie neuer Text ausgezeichnet wird. Ohne andere Definition gilt, dass der Text farbig oder je nach Option `markup` bzw. `addedmarkup` erscheint.

Werte für Definition: beliebige  $\TeX$ -Befehle, der neue Text wird mit „#1“ eingesetzt.

*Aufruf*

```
\setaddedmarkup{\definition}
```

*Beispiele*

```
\setaddedmarkup{\emph{#1}}
\setaddedmarkup{+++ : #1}
```

### 4.6.2 \setdeletedmarkup

`\setdeletedmarkup`

Der Befehl `\setdeletedmarkup` legt fest, wie gelöschter Text ausgezeichnet wird. Ohne andere Definition gilt, dass der Text durchgestrichen wird oder je nach Option `markup` bzw. `deletedmarkup` erscheint.

Werte für Definition: beliebige  $\TeX$ -Befehle, der gelöschte Text wird mit „#1“ eingesetzt.

*Aufruf*

```
\setdeletedmarkup{\definition}
```

*Beispiele*

```
\setdeletedmarkup{\emph{#1}}
\setdeletedmarkup{--- : #1}
```

### 4.6.3 `\sethighlightmarkup`

`\sethighlightmarkup`

Der Befehl `\sethighlightmarkup` legt fest, wie hervorgehobene Texte gesetzt werden. Ohne andere Definition gilt, dass die Hervorhebung über die Hintergrundfarbe erfolgt oder je nach Option `markup` bzw. `commentmarkup` erscheint.

Werte für Definition: beliebige  $\TeX$ -Befehle, der hervorgehobene Text wird mit „#1“ eingesetzt. Die Autorenfarbe kann mit „authorcolor“ genutzt werden.

*Aufruf*

```
\sethighlightmarkup{<definition>}
```

*Beispiele*

```
\sethighlightmarkup{\emph{#1}}
```

```
\sethighlightmarkup{\ifthenelse{\isColored}{\color{authorcolor!60}}{!!: #1}}
```

### 4.6.4 `\setcommentmarkup`

`\setcommentmarkup`

Der Befehl `\setcommentmarkup` legt fest, wie Kommentare gesetzt werden. Ohne andere Definition gilt, dass Kommentare im Rand oder je nach Option `markup` bzw. `commentmarkup` erscheint.

Werte für Definition: beliebige  $\TeX$ -Befehle, der Kommentar wird mit „#1“ eingesetzt, die Autor-ID mit „#2“ und die Autor-Ausgabe (ID oder Name) mit „#3“. Die Autorenfarbe kann mit „authorcolor“ genutzt werden und die Kommentaranzahl mit „authorcommentcount“.

*Aufruf*

```
\setcommentmarkup{<definition>}
```

*Beispiele*

```
\setcommentmarkup{-- #1 --}
```

```
\setcommentmarkup{\ifthenelse{\isColored}{\color{authorcolor}}{}}{#1}
```

```
\setcommentmarkup{\ifthenelse{\isAnonymous{#2}}{\textbf{#3: }}{#1}}
```

```
\setcommentmarkup{[\arabic{authorcommentcount}] #1}
```

### 4.6.5 `\setauthormarkup`

`\setauthormarkup`

Der Befehl `\setauthormarkup` legt fest, wie der Autortext im Text angezeigt wird. Ohne andere Definition gilt, dass der Autor hochgestellt erscheint.

Werte für Definition: beliebige  $\TeX$ -Befehle, der Autortext wird mit „#1“ eingesetzt.

*Aufruf*

```
\setauthormarkup{⟨definition⟩}
```

*Beispiele*

```
\setauthormarkup{(#1)}
```

```
\setauthormarkup{(#1)~---~}
```

```
\setauthormarkup{\marginpar{#1}}
```

**4.6.6 \setauthormarkupposition**

`\setauthormarkupposition`

Der Befehl `\setauthormarkupposition` legt fest, auf welcher Seite der Autor im Text angezeigt wird. Ohne andere Definition gilt, dass der Autor rechts von den Änderungen erscheint.

Mögliche Werte: *left* == links von den Änderungen; alles andere: rechts

*Aufruf*

```
\setauthormarkupposition{⟨position⟩}
```

*Beispiele*

```
\setauthormarkupposition{left}
```

**4.6.7 \setauthormarkuptext**

`\setauthormarkuptext`

Der Befehl `\setauthormarkuptext` legt fest, welche Information des Autors im Text angezeigt wird. Ohne andere Definition gilt, dass die Autor-ID genutzt wird.

Mögliche Werte: *name* == Autorennamen; alles andere: Autor-ID

*Aufruf*

```
\setauthormarkuptext{⟨text⟩}
```

*Beispiele*

```
\setauthormarkuptext{name}
```

**4.6.8 \settruncatewidth**

`\settruncatewidth`

Der Befehl `\settruncatewidth` legt die Breite der Textkürzung in der Änderungsliste fest. Die Standardbreite ist `0.6\textwidth`.

```
\settruncatewidth{<length>}
```

```
\settruncatewidth{5cm}  
\settruncatewidth{.3\textwidth}
```

#### 4.6.9 `\setsummarywidth`

```
\setsummarywidth
```

Der Befehl `\setsummarywidth` legt die Breite der Änderungsliste mit Stil `summary` bzw. `compactsummary` fest. Die Standardbreite ist `0.3\textwidth`.

```
\setsummarywidth{<length>}
```

```
\setsummarywidth{3cm}  
\setsummarywidth{.5\textwidth}
```

#### 4.6.10 `\setsummarytewidth`

```
\setsummarytewidth
```

Der Befehl `\setsummarytewidth` legt die Breite der Änderungsliste mit Stil `summary` bzw. `compactsummary` anhand der Breite des übergebenen Texts fest.

```
\setsummarytewidth{<text>}
```

```
\setsummarytewidth{Highlighted \qquad}  
\setsummarytewidth{The longest text you can imagine for the summary.}
```

#### 4.6.11 `\setsocextension`

```
\setsocextension
```

Der Befehl `\setsocextension` legt das Suffix der Hilfsdatei für die Änderungszusammenfassung (soc-Datei<sup>3</sup>) fest. Ohne andere Definition gilt das Suffix „soc“. Im unten angegebenen Beispiel würde für „foo.tex“ eine Hilfsdatei erzeugt werden, die „foo.changes“ statt des Standardnamens „foo.soc“ hieße.

*Aufruf*

```
\setsocextension{<extension>}
```

*Beispiele*

```
\setsocextension{changes}
```

---

<sup>3</sup> „soc“ steht dabei für „summary of changes“.

## 4.7 Benötigte Pakete

Das *changes*-Paket bindet bereits Pakete ein, die für die Funktion des Pakets notwendig sind. Eine genauere Beschreibung der einzelnen Pakete ist in der Dokumentation der Pakete selbst zu finden.

Die folgenden Pakete sind zwingend notwendig und müssen für die Nutzung des *changes*-Pakets installiert sein:

*xifthen*

stellt eine verbesserte *if*-Abfrage sowie eine *while*-Schleife zur Verfügung

*xkeyval*

Eingabe von Optionen mit Werteübergabe

*xstring*

verbesserte Stringoperationen

Die folgenden Pakete sind manchmal notwendig und müssen installiert sein, wenn sie über die entsprechende Option genutzt werden:

*pdfcolmk*

wird geladen, wenn farbiger Text genutzt wird (default Markup); löst das Problem farbigen Texts über Seitenumbrüche hinweg (bei *pdflatex*)

*todonotes*

wird geladen, wenn Kommentare als ToDo-Notizen gesetzt werden (default Markup)

*ulem*

wird geladen, wenn Text durchgestrichen oder ausge-x-t wird (default Markup)

*xcolor*

wird geladen, wenn farbiger Text genutzt wird (default Markup)



## 5 Autorinnen und Autoren

Am *changes*-Paket haben mehrere Autorinnen und Autoren mitgewirkt. Viele Probleme wurden in `de.comp.text.tex` gelöst oder deren Lösung durch Lösungsansätzen inspiriert. Danke.

Die Autorinnen und Autoren sind in alphabetischer Reihenfolge:

- Chiaradonna, Silvano
- Fischer, Ulrike
- Giovannini, Daniele
- Kleinod, Ekkart
- Mittelbach, Frank
- Voss, Herbert
- Wölfel, Philipp
- Wolter, Steve

## 6 Versionen

Für eine Liste der verfügbaren Versionen und deren Änderungen gehen Sie bitte zu

<https://gitlab.com/ekleinod/changes/blob/master/changelog.md>

Dort sind auch die bereits implementierten aber noch nicht veröffentlichten Änderungen verzeichnet.

Wenn Sie an geplanten, zukünftigen Änderungen interessiert sind, finden Sie diese unter

<https://gitlab.com/ekleinod/changes/milestones>

## 7 Weitergabe, Copyright, Lizenz

Copyright 2007-2018 Ekkart Kleinod (ekleinod@edgesoft.de)

Dieses Paket darf unter der „ $\text{\LaTeX}$  Project Public License“ Version 1.3 oder jeder späteren Version weitergegeben und/oder geändert werden. Die neueste Version dieser Lizenz steht auf <http://www.latex-project.org/lppl.txt> Version 1.3 und spätere Versionen sind Teil aller  $\text{\LaTeX}$ -Distributionen ab Version 2005/12/01.

Dieses Paket besitzt den Status „maintained“ (verwaltet). Der aktuelle Verwalter dieses Pakets ist Ekkart Kleinod.

Dieses Paket besteht aus den Dateien

```
source/latex/changes/changes.drv
source/latex/changes/changes.dtx
source/latex/changes/changes.ins
source/latex/changes/examples.dtx
source/latex/changes/README
source/latex/changes/userdoc/*.tex
scripts/changes/delcmdchanges.bash
```

und den generierten Dateien

```
doc/latex/changes/changes.english.pdf
doc/latex/changes/changes.english.withcode.pdf
doc/latex/changes/changes.ngerman.pdf
doc/latex/changes/examples/changes.example.*.tex
doc/latex/changes/examples/changes.example.*.pdf
tex/latex/changes/changes.sty
```