

Das *changes*-Paket

Manuelle Änderungsmarkierung – Version 0.5.3

Ekkart Kleinod
ekkart@ekkart.de

22. November 2010

Inhaltsverzeichnis

1	Einleitung	3
2	Die Benutzerschnittstelle des <i>changes</i>-Pakets	3
2.1	Benötigte Pakete	3
2.2	Optionen	4
2.3	Neue Befehle	4
2.3.1	Änderungsmanagement	5
2.3.2	Nutzerdefinierte Anpassungen	6
3	Der kommentierte Quellcode	7
3.1	Paketinfo und Optionen	7
3.2	Einstellungen	9
3.3	Autorenverwaltung	10
3.4	Änderungsbefehle	11
3.5	Änderungsliste	13
4	Beispieldateien	15
4.1	Rudimentär ohne <i>babel</i>	15
4.2	Rudimentär mit <i>babel</i>	15
4.3	Erweiterte Einstellungen	16
5	Autoren	17
6	Versionen	18
6.1	TODO/bekannte Fehler	18
6.2	Version 0.5.3	18
6.3	Version 0.5.2	18
6.4	Version 0.5.1	18
6.5	Version 0.5	19
6.6	Version 0.4	19
6.7	Version 0.3	19
6.8	Version 0.2	20
6.9	Version 0.1	20
7	Weitergabe, Copyright, Lizenz	20

1 Einleitung

Dieses Paket dient dazu, manuelle Änderungsmarkierung anzubieten.

Ein offenes Ohr für Verbesserungsvorschläge oder Kritik habe ich per Mail immer.

Diejenigen, die den kommentierten Quellcode des Pakets lesen wollen, kommentieren bitte die Anweisung `\OnlyDescription` in der Datei `changes.drv` aus und übersetzen diese.

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. Das Paket ermöglicht die freie Definition von Autoren und deren zugeordneten Farben. Es erlaubt zusätzlich die Definition des Autor- und Anmerkungsmarkups.

2 Die Benutzerschnittstelle des *changes*-Pakets

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

```
\usepackage{changes}
```

Die Paketoptionen werden in [Unterabschnitt 2.2](#) erläutert.

2.1 Benötigte Pakete

Das *changes*-Paket bindet bereits Pakete ein, die für das Paket notwendig sind. Die Pakete werden im folgenden kurz vorgestellt und müssen für die Nutzung des *changes*-Pakets zur Verfügung stehen, sonst gibt es Fehlermeldungen.

Eine genauere Beschreibung der einzelnen Pakete ist in der Dokumentation der Pakete selbst zu finden.

ifthen stellt eine verbesserte `if`-Abfrage sowie eine `while`-Schleife zur Verfügung

pdfcolmk löst das Problem farbigen Texts über Seitenümbrüche hinweg (bei `pdflatex`)

ulem Durchstreichen von Texten

xcolor farbige Markierung von Texten

xkeyval Eingabe von Optionen mit Werteübergabe

2.2 Optionen

draft

Die *draft*-Option bewirkt, dass alle Änderungen markiert werden. Die Änderungsliste kann durch `\listofchanges` ausgegeben werden. Ohne Optionsangabe wird *draft* automatisch eingestellt.

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}
```

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}
```

ulem

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

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

xcolor

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

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

2.3 Neue Befehle

Dieser Abschnitt führt alle neuen Befehle auf und erläutert sie.

2.3.1 Änderungsmanagement

`\added`

Der Befehl `\added` markiert zugefügten Text. Der neue Text wird als notwendiges Argument in geschweiften Klammern übergeben. Optional können eine Autoren-ID sowie eine Anmerkung übergeben werden. Die Autoren-ID muss mit einer mit dem `\definechangesauthor`-Befehl definierten ID übereinstimmen. Soll nur eine Anmerkung (ohne Autor) eingegeben werden, so ist statt des Autors ein leeres Argument zu übergeben.

Definition `\added[⟨Autor-ID⟩][⟨Anmerkung⟩]{⟨neuer Text⟩}`

Beispiel Das ist `\added[EK]{neuer}` Text.
 Das ist `neuerEK` Text.
 Das ist `\added[][anonym]{neuer}` Text.
 Das ist `neuer1` Text.

`\deleted`

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

Definition `\deleted[⟨Autor-ID⟩][⟨Anmerkung⟩]{⟨gelöschter Text⟩}`

Beispiel Das ist `\deleted[][obsolet]{schlechter}` Text.
 Das ist `schlechter2` Text.

`\replaced`

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

Definition `\replaced[⟨Autor-ID⟩][⟨Anmerkung⟩]{⟨neuer Text⟩}{⟨alter Text⟩}`

Beispiel Das ist `\replaced[EK]{schöner}{schlechter}` Text.
 Das ist `schönerschlechterEK` Text.

`\listofchanges`

Der Befehl `\listofchanges` gibt eine Liste 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.

Definition `\listofchanges`

2.3.2 Nutzerdefinierte Anpassungen

`\definechangesauthor`

Der Befehl `\definechangesauthor` definiert einen neuen Autor für Änderungen. Es muss eine eindeutige Autor-ID und dessen Farbe angegeben werden. Die Autor-ID darf keine Sonder- oder Leerzeichen enthalten. Ein Autorenname kann optional angegeben werden, er wird in der Änderungsliste benutzt.

Definition `\definechangesauthor[<Autor-Name>]{<Autor-ID>}{<Farbe>}`

Beispiel `\definechangesauthor{EK}{orange}`
`\definechangesauthor[Ekkart Kleinod]{EK}{orange}`

`\setlocextension`

Der Befehl `\setlocextension` legt das Suffix der Hilfsdatei für die Änderungsliste fest. Ohne andere Definition gilt das Suffix „loc“. Das Beispiel würde für „foo.tex“ Hilfsdateien erzeugen, die „foo.changes“ heißen.

Definition `\setlocextension{<extension>}`

Beispiel `\setlocextension{changes}`

`\setauthormarkup`

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

Werte für Position (optional): `left` = links von den Änderungen; alles andere: rechts
Werte für Definition: beliebige \TeX -Befehle, der Autorenname wird mit „#1“ gekennzeichnet.

Definition `\setauthormarkup[<position>]{<definition>}`

Beispiel `\setauthormarkup{(#1)}`
`\setauthormarkup[left]{(#1)~---}`
`\setauthormarkup{\marginpar{#1}}`
`\setauthormarkup[right]{}`

```
\setremarkmarkup
```

Der Befehl `\setremarkmarkup` legt fest, wie die Anmerkungen im Text angezeigt werden. Ohne andere Definition gilt, dass die Anmerkungen als Fußnote mit farbigem Text gesetzt werden.

Werte für Definition: beliebige \TeX -Befehle, die Autor-ID wird mit „#1“ benutzt, der Anmerkungstext mit „#2“. Über die Autor-ID kann mit `Changes@Color#1` die Farbe des Autors benutzt werden.

Definition `\setremarkmarkup{<definition>}`

Beispiel `\setremarkmarkup{(#2:#1)}`
`\setremarkmarkup{\footnote{#1:\textcolor{Changes@Color#1}{#2}}}`

3 Der kommentierte Quellcode

3.1 Paketinfo und Optionen

```
1 <*changes>
```

Zunächst legen wir die benötigte \TeX -Version auf $\text{\TeX}2_{\epsilon}$ fest und geben Name, Datum und Version des Pakets zurück. Außerdem werden einige Ausgaben auf die Konsole gegeben.

```
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{changes}
4 [2010/11/22 v0.5.3 changes-Paket]
5 \typeout{*** changes-Paket 2010/11/22 v0.5.3 ***}
```

Das Paket `xkeyval` ist für die Eingabe von Optionen mit Werteübergabe zuständig.

```
6 \RequirePackage{xkeyval}
```

Das Paket `ifthen` stellt eine verbesserte `if`-Abfrage sowie eine `while`-Schleife zur Verfügung.

```
7 \RequirePackage{ifthen}
```

Optionen

draft-Option, default

```
8 \newboolean{Changes@optiondraft}
9 \setboolean{Changes@optiondraft}{true}
10 \DeclareOptionX{draft}{
11 \setboolean{Changes@optiondraft}{true}}
```

```
12 \typeout{changes-option '\CurrentOption'}
13 }
```

final-Option

```
14 \DeclareOptionX{final}{
15 \setboolean{Changes@optiondraft}{false}
16 \typeout{changes-option '\CurrentOption'}
17 }
```

Optionen für das *ulem*-Paket

```
18 \DeclareOptionX{ulem}{
19 \typeout{ulem-option '#1', passed to package ulem}
20 \PassOptionsToPackage{#1}{ulem}
21 }
```

Optionen für das *xcolor*-Paket

```
22 \DeclareOptionX{xcolor}{
23 \typeout{xcolor-option '#1', passed to package xcolor}
24 \PassOptionsToPackage{#1}{xcolor}
25 }
```

Unbekannte Optionen

```
26 \DeclareOptionX*{
27 \PackageWarning{Unknown changes-option '\CurrentOption'}
28 }
```

Ausführung des Optionenbearbeitens.

```
29 \ProcessOptionsX*\relax
```

Das Paket *xcolor* stellt die Färbung von Texten zur Verfügung.

```
30 \RequirePackage{xcolor}
```

Das Paket *pdfcolmk* löst das Problem farbigen Texts über Seitenumbrüche hinweg (muss nach *xcolor* geladen werden).

```
31 \RequirePackage{pdfcolmk}
```

Das Paket *ulem* stellt Kommandos für das Durchstreichen von Text zur Verfügung.

```
32 \RequirePackage{ulem}
```

3.2 Einstellungen

Bezeichnungen.

```

33 \ifthenelse{\isundefined{\addto}}
34 {
35 \def\listchangesname{"Anderungen}
36 \def\changesaddname{Eingef\ugt}
37 \def\changesdeletename{Gel\oscht}
38 \def\changesreplacename{Ersetzt}
39 \def\changesauthorname{Autor}
40 \def\changesanonymousname{Anonym}
41 \def\changesnoloc{"Anderungsliste erst nach dem n\achen Durchlauf verf\ugbar.}
42 }{
43 \addto\captionsgerman{\def\listchangesname{"Anderungen}}
44 \addto\captionsgerman{\def\listchangesname{"Anderungen}}
45 \addto\captionsenglish{\def\listchangesname{Changes}}
46
47 \addto\captionsgerman{\def\changesaddname{Eingef\ugt}}
48 \addto\captionsgerman{\def\changesaddname{Eingef\ugt}}
49 \addto\captionsenglish{\def\changesaddname{Added}}
50
51 \addto\captionsgerman{\def\changesdeletename{Gel\oscht}}
52 \addto\captionsgerman{\def\changesdeletename{Gel\oscht}}
53 \addto\captionsenglish{\def\changesdeletename{Deleted}}
54
55 \addto\captionsgerman{\def\changesreplacename{Ersetzt}}
56 \addto\captionsgerman{\def\changesreplacename{Ersetzt}}
57 \addto\captionsenglish{\def\changesreplacename{Replaced}}
58
59 \addto\captionsgerman{\def\changesauthorname{Autor}}
60 \addto\captionsgerman{\def\changesauthorname{Autor}}
61 \addto\captionsenglish{\def\changesauthorname{Author}}
62
63 \addto\captionsgerman{\def\changesanonymousname{Anonym}}
64 \addto\captionsgerman{\def\changesanonymousname{Anonym}}
65 \addto\captionsenglish{\def\changesanonymousname{anonymous}}
66
67 \addto\captionsgerman{\def\changesnoloc{"Anderungsliste erst nach dem n\achen \LaTeX-Lauf}}
68 \addto\captionsgerman{\def\changesnoloc{"Anderungsliste erst nach dem n\achen Durchlauf v}}
69 \addto\captionsenglish{\def\changesnoloc{List of changes is available after the next \LaTeX\ l}}
70 }

```

Dateiendung.

```
71 \newcommand{\Changes@extension}{loc}
```

`\setlocextension` Dieser Befehl überschreibt die Default-Dateiendung. Argument: neue Endung

```
72 \newcommand{\setlocextension}[1]{
73 \renewcommand{\Changes@extension}{#1}
74 }
```

3.3 Autorenverwaltung

Autoren-Zähler.

```
75 \newcounter{Changes@AuthorCount}
76 \setcounter{Changes@AuthorCount}{0}
77 \newcounter{Changes@iAuthor}
```

`definechangesauthor` Dieser Befehl definiert einen neuen Autor. Argumente: Autor-ID und Farbe des Autors. Optionales Argument: Autorenname.

Autor-ID, Name und Farbe mit benannten Variablen merken. Pro Autor eigenen Zähler und Farben definieren.

```
78 \newcommand{\definechangesauthor}[3][\@empty]{
79 \stepcounter{Changes@AuthorCount}
80 \@namedef{Changes@Author\theChanges@AuthorCount}{#2}
81 \@namedef{Changes@AuthorName#2}{#1}
82 \@namedef{Changes@AuthorColor#2}{#3}
83 \newcounter{Changes@AddCount#2}
84 \newcounter{Changes@DeleteCount#2}
85 \newcounter{Changes@ReplaceCount#2}
86 \colorlet{Changes@Color#2}{#3}
87 }
```

Default-Autor definieren (anonym).

```
88 \definechangesauthor{\@empty}{blue}
```

Position der Autormarkierung bzgl. eingegebenem Text. `true` = links, `false` = rechts (default).

```
89 \newboolean{Changes@AuthorPositionLeft}
90 \setboolean{Changes@AuthorPositionLeft}{false}
```

`\Changes@AuthorMark` Ausgabe des Autors bei eingegebenem Text. Default: hochgestellt.

```
91 \newcommand{\Changes@AuthorMark}[1]{%
92 \textsuperscript{#1}%
93 }
```

`\setauthormarkup` Neudefinition der Autorenausgabe.

```

94 \newcommand{\setauthormarkup}[2][left]{%
95 \ifthenelse{\equal{#1}{left}}
96 {\setboolean{Changes@AuthorPositionLeft}{true}}
97 {\setboolean{Changes@AuthorPositionLeft}{false}}
98 \renewcommand{\Changes@AuthorMark}[1]{#2}
99 }

```

`\Changes@Remark` Ausgabe der Anmerkung. Default: als Fußnote.

```

100 \newcommand{\Changes@Remark}[2]{%
101 \footnote{\textcolor{Changes@Color#1}{#2}}%
102 }

```

`\setremarkmarkup` Neudefinition der Anmerkungen.

```

103 \newcommand{\setremarkmarkup}[1]{%
104 \renewcommand{\Changes@Remark}[2]{#1}%
105 }

```

3.4 Änderungsbefehle

Temporärer Speicher, um Parameterübergabe zwischen den geschachtelten Kommandos zu erledigen.

```

106 \newcommand{\Changes@temp}{\@empty}

```

`\added` Dieser Befehl formatiert den Text als neuen Text. Die komplizierte Konstruktion ist nötig, um ein Kommando mit zwei optionalen Argumenten zu definieren.

Argumente: Autor-ID (opt.), Erläuterung (opt.) und neuer Text

```

107 \newcommand{\added}[1][\@empty]{%
108 \renewcommand{\Changes@temp}{#1}%
109 \Changes@added%
110 }
111 \newcommand{\Changes@added}[2][\@empty]{%
112 \ifthenelse{\boolean{Changes@optiondraft}}%
113 {%
114 \textcolor{Changes@Color\Changes@temp}{%
115 \ifthenelse{\boolean{Changes@AuthorPositionLeft}}%
116 {\ifthenelse{\equal{\@empty}{\Changes@temp}}%
117 }{\Changes@AuthorMark{\Changes@temp}}%
118 }{}%
119 #2%
120 \ifthenelse{\not\boolean{Changes@AuthorPositionLeft}}%

```

```
121 {\ifthenelse{\equal{\@empty}{\Changes@temp}}}%
122 }{\Changes@AuthorMark{\Changes@temp}}}%
123 }{}%
124 \ifthenelse{\equal{\@empty}{#1}}%
125 }{\Changes@Remark{\Changes@temp}{#1}}%
126 }%
127 \stepcounter{Changes@AddCount\Changes@temp}%
128 }{#2}%
129 }
```

`\deleted` Dieser Befehl formatiert den Text als gelöschten Text. Die komplizierte Konstruktion ist nötig, um ein Kommando mit zwei optionalen Argumenten zu definieren.

Der `final`-Teil ist einem Tip von `de.comp.text.tex` entnommen. Er löst das Problem des zusätzlichen Leerraums bei leeren Kommandos.

Argumente: Autor-ID (opt.), Erläuterung (opt.) und gelöschter Text

```
130 \newcommand{\deleted}[1][\@empty]{%
131 \renewcommand{\Changes@temp}{#1}%
132 \Changes@deleted%
133 }
134 \newcommand{\Changes@deleted}[2][\@empty]{%
135 \ifthenelse{\boolean{Changes@optiondraft}}%
136 {%
137 \textcolor{Changes@Color\Changes@temp}{%
138 \ifthenelse{\boolean{Changes@AuthorPositionLeft}}%
139 }{\ifthenelse{\equal{\@empty}{\Changes@temp}}%
140 }{\Changes@AuthorMark{\Changes@temp}}%
141 }{}%
142 \sout{#2}%
143 \ifthenelse{\not\boolean{Changes@AuthorPositionLeft}}%
144 }{\ifthenelse{\equal{\@empty}{\Changes@temp}}%
145 }{\Changes@AuthorMark{\Changes@temp}}%
146 }{}%
147 \ifthenelse{\equal{\@empty}{#1}}%
148 }{\Changes@Remark{\Changes@temp}{#1}}%
149 }%
150 \stepcounter{Changes@DeleteCount\Changes@temp}%
151 }{\@bsphack \expandafter \@esphack}%
152 }
```

`\replaced` Dieser Befehl formatiert den Text als ersetzten Text. Die komplizierte Konstruktion ist nötig, um ein Kommando mit zwei optionalen Argumenten zu definieren.

Argumente: Autor-ID (opt.), Erläuterung (opt.), neuer Text und alter Text

```
153 \newcommand{\replaced}[1][\@empty]{%
154 \renewcommand{\Changes@temp}{#1}%
```

```

155 \Changes@replaced%
156 }
157 \newcommand{\Changes@replaced}[3][\@empty]{%
158 \ifthenelse{\boolean{Changes@optiondraft}}{%
159 {%
160 \textcolor{Changes@Color\Changes@temp}{%
161 \ifthenelse{\boolean{Changes@AuthorPositionLeft}}{%
162 {\ifthenelse{equal{\@empty}{\Changes@temp}}%
163 }{\Changes@AuthorMark{\Changes@temp}}%
164 }{}%
165 #2%
166 \sout{#3}%
167 \ifthenelse{\not\boolean{Changes@AuthorPositionLeft}}%
168 {\ifthenelse{equal{\@empty}{\Changes@temp}}%
169 }{\Changes@AuthorMark{\Changes@temp}}%
170 }{}%
171 \ifthenelse{equal{\@empty}{#1}}%
172 }{\Changes@Remark{\Changes@temp}{#1}}%
173 }%
174 \stepcounter{Changes@ReplaceCount\Changes@temp}%
175 }{#2}%
176 }

```

3.5 Änderungsliste

`\changes@chopline` Hilfsbefehl zum Einlesen der loc-Dateien.

```

177 \def\changes@chopline#1;#2;#3;#4;#5;#6 \{\
178 \def\Changes@InID{#1}
179 \def\Changes@InColor{#2}
180 \def\Changes@InName{#3}
181 \def\Changes@InAdded{#4}
182 \def\Changes@InDeleted{#5}
183 \def\Changes@InReplaced{#6}
184 }

```

`\listofchanges` Dieser Befehl gibt eine Liste der Änderungen nach Autoren aus. Die Werte werden aus der loc-Datei gelesen, wenn diese existiert.

```

185 \newcommand{\listofchanges}{%
186 \ifthenelse{\boolean{Changes@optiondraft}}
187 {
188 \section*{\listchangesname}
189 \IfFileExists{\jobname.\Changes@extension}
190 {
191 \newboolean{Changes@MoreLines}
192 \setboolean{Changes@MoreLines}{true}
193 \newread\Changes@InFile

```

```

194 \openin\Changes@InFile = \jobname.\Changes@extension
195 \whiledo{\boolean{Changes@MoreLines}}{
196 \read\Changes@InFile to \Changes@Line
197 \ifeof\Changes@InFile
198 \setboolean{Changes@MoreLines}{false}
199 \else
200 \expandafter\changes@chopline\Changes@Line\
201 \begin{tabbing}
202 mm\=mmmmmm\=\kill
203 \ifthenelse{\equal{\Changes@InID}{}}
204 {\textcolor{\Changes@InColor}{\changesauthorname: \changesanonymousname}}\}
205 {\textcolor{\Changes@InColor}{\changesauthorname: \Changes@InID}}
206 \ifthenelse{\equal{\Changes@InName}{}}{
207 { \textcolor{\Changes@InColor}{(\Changes@InName)}}
208 \}
209 }
210 \>\changesaddname: \>\Changes@InAdded\
211 \>\changesdeletename: \>\Changes@InDeleted\
212 \>\changesreplacename: \>\Changes@InReplaced\
213 \end{tabbing}
214 \fi
215 }
216 \closein\Changes@InFile
217 }{
218 \emph{\changesnoloc}
219 }
220 }{}
221 }

```

Ausgabe der Änderungen in loc-Datei, dazu Datei öffnen, Werte herschreiben, Datei schließen. Die Änderungen werden als \LaTeX -formatierter Text ausgegeben, der mit `\input` eingelesen werden kann.

Zeilenformat: ID;Farbe;Name;Zugefügte;Gelöschte;Geänderte

```

222 \AtEndDocument{
223 \newwrite\Changes@OutFile
224 \immediate\openout\Changes@OutFile = \jobname.\Changes@extension
225 \setcounter{Changes@iAuthor}{0}
226 \whiledo{\value{Changes@iAuthor} < \value{Changes@AuthorCount}}{
227 \stepcounter{Changes@iAuthor}
228 \def\Changes@ID{\@nameuse{Changes@Author\theChanges@iAuthor}}
229 \immediate\write\Changes@OutFile{\Changes@ID;%
230 \@nameuse{Changes@AuthorColor\Changes@ID};%
231 \@nameuse{Changes@AuthorName\Changes@ID};%
232 \the\value{Changes@AddCount\Changes@ID};%
233 \the\value{Changes@DeleteCount\Changes@ID};%
234 \the\value{Changes@ReplaceCount\Changes@ID}}
235 }

```

```

236 \closeout\Changes@OutFile
237 }

238 </changes>

```

4 Beispieldateien

4.1 Rudimentär ohne *babel*

```

239 <*example : 1>
240 \documentclass{article}
241
242 % draft = Ausgabe der Änderungen
243 \usepackage[draft]{changes}
244
245 \definechangesauthor{EK}{orange}
246
247 \begin{document}
248
249 Dieses Beispiel zeigt die rudiment\aren Funktionen.
250
251 \listofchanges
252
253 Dieser Text ist nicht modifiziert.
254
255 Hier \added{f\"uge} ich Text anonym \added{ein}.
256 Hier \deleted{l\"osche} ich anonym Text.
257 Und an dieser Stelle \replaced{"andere}{alt} ich anonym Text.
258 Anonyme \deleted[][Ja!]{L\"oschung} mit Anmerkung.
259
260 Hier \added[EK]{f\"uge} ich Text als Autor "EK" \added[EK]{ein}.
261 Hier f\"uge ich \added[EK][Weil ich es kann.]{Text}
262 als Autor "EK" mit Begr\"undung ein.
263
264 \end{document}
265 </example : 1>

```

4.2 Rudimentär mit *babel*

```

266 <*example : 2>
267 \documentclass[ngerman]{article}
268
269 \usepackage{babel}
270 \usepackage[utf8]{inputenc}
271 \RequirePackage[T1]{fontenc}
272
273 % draft = Ausgabe der Änderungen

```

```
274 \usepackage[draft]{changes}
275
276 \definechangesauthor{EK}{orange}
277
278 \begin{document}
279
280 Dieses Beispiel zeigt die rudimentären Funktionen
281 unter Einbeziehung des babel-Pakets.
282
283 \listofchanges
284
285 Dieser Text ist nicht modifiziert.
286
287 Hier \added{füge} ich Text anonym \added{ein}.
288 Hier \deleted{lösche} ich anonym Text.
289 Und an dieser Stelle \replaced{ändere}{alt} ich anonym Text.
290 Anonyme \deleted[][Ja!]{Löschung} mit Anmerkung.
291
292 Hier \added[EK]{füge} ich Text als Autor "EK" \added[EK]{ein}.
293 Hier füge ich \added[EK][Weil ich es kann.]{Text}
294 als Autor "EK" mit Begründung ein.
295
296 \end{document}
297 \end{example : 2}
```

4.3 Erweiterte Einstellungen

```
298 \example : 3
299 \documentclass[ngerman]{article}
300
301 \usepackage{babel}
302 \usepackage[utf8]{inputenc}
303 \RequirePackage[T1]{fontenc}
304
305 % draft = Ausgabe der Änderungen
306 \usepackage[draft]{changes}
307
308 \definechangesauthor[Ekkart Kleinod]{EK}{orange}
309 \definechangesauthor[Test]{green}
310
311 \setlocextension{changes}
312 \setauthormarkup[left]{(#1)~---~}
313 \setremarkmarkup{(#2:#1)}
314
315 \begin{document}
316
317 Dieses Beispiel zeigt die erweiterten Funktionen.
318 Es setzt die Erweiterung der Hilfsdatei auf \texttt{changes}.
319 Ein Autorname wird angegeben.
320 Die Autorenmarkierung wird links gesetzt.
```

```

321 Sie besteht aus dem eingeklammerten Autorennamen,
322 der durch Leerzeichen und einen Gedankenstrich
323 von der Änderung abgesetzt ist.
324 Die Anmerkung wird in Klammern gesetzt, der Autorname dahinter.
325
326 \listofchanges
327
328 Dieser Text ist nicht modifiziert.
329
330 Hier \added{füge} ich Text anonym \added{ein}.
331 Hier \deleted{lösche} ich anonym Text.
332 Und an dieser Stelle \replaced{ändere}{alt} ich anonym Text.
333 Anonyme \deleted[][Ja!]{Löschung} mit Anmerkung.
334
335 Hier \added[EK]{füge} ich Text als Autor "'EK"' \added[EK]{ein}.
336 Hier füge ich \added[EK][Weil ich es kann.]{Text}
337 als Autor "'EK"' mit Begründung ein.
338
339 Hier \deleted[Test][Weil ich es will.]{lösche} ich Text
340 als Autor "'Text"' .
341
342 Test von Zeilenumbrüchen.
343 \added{eingefügt eingefügt eingefügt eingefügt eingefügt
344 eingefügt eingefügt eingefügt eingefügt eingefügt eingefügt
345 eingefügt eingefügt eingefügt eingefügt eingefügt.}
346 \deleted{gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht
347 gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht
348 gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht.}
349 \replaced{eingefügt eingefügt eingefügt eingefügt eingefügt
350 eingefügt eingefügt eingefügt eingefügt eingefügt eingefügt
351 eingefügt eingefügt eingefügt eingefügt eingefügt.}
352 {gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht
353 gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht
354 gelöscht gelöscht gelöscht gelöscht gelöscht gelöscht.}
355
356 \end{document}
357 </example : 3>

```

5 Autoren

Am *changes*-Paket haben mehrere Autoren mitgewirkt. Dies sind in alphabetischer Reihenfolge:

- Kleinod, Ekkart
- Wolter, Steve

6 Versionen

6.1 TODO/bekannte Fehler

- Ant-Task zur Entfernung aller *changes*-Befehle
- Bild/Tabellenauszeichnung
- Änderungsumgebungen für längere Passagen
- bei fehlenden T1-Fonts (Umlautkodierung) und *inputenc* gibt es Fehler
- evtl. zweite Klasse von Änderungen: nicht so wichtige (extra ausblendbar), vielleicht über Stern-Variante?

6.2 Version 0.5.3

Datum: 22. 11. 2010

- Dokumentoptionen von `\documentclass` werden ebenfalls genutzt (Vorschlag und Code von Steve Wolter)

6.3 Version 0.5.2

Datum: 10. 10. 2007

- Paketoptionen der Pakete *pdfcolmk*, *ulem*, and *xcolor* werden weitergeleitet

6.4 Version 0.5.1

Datum: 27. 08. 2007

- gelöschter Text wieder durchgestrichen, Paket *ulem* funktioniert; ausgrauen hat nicht funktioniert

6.5 Version 0.5

Datum: 26.08.2007

- keine Nutzung des *arrayjob*-Pakets mehr, dadurch Fehler im Zusammenspiel mit *array* behoben
- auf UTF-8-encoding umgestellt
- keine Nutzung des *soul*-Pakets mehr, dadurch Fehler im Zusammenspiel UTF-8-encoding behoben
- gelöschter Text durch grauen Hintergrund visualisiert (es gibt bisher kein ordentliches Durchstreichen bei UTF-8-Nutzung)
- neues optionales Argument für Autorennamen
- farbige Liste der Änderungen
- loc-Format geändert
- englische Doku verbessert

6.6 Version 0.4

Datum: 24.01.2007

- *pdfcolmk* eingebunden, um Problem mit farbigem Text bei Seitenumbrüchen zu lösen
- `\setremarkmarkup` um Autor-ID erweitert, um Anmerkung farbig setzen zu können
- Anmerkungen werden in der Fußnote farbig gesetzt
- erste Version für das CTAN

6.7 Version 0.3

Datum: 22.01.2007

- englische Nutzerdokumentation
- Befehl `\changed` ersetzt durch `\replaced`
- verbesserte `final`-Option: kein zusätzlicher Leerraum

6.8 Version 0.2

Datum: 17. 01. 2007

- Bezeichnungen auch bei fehlendem *babel*-Paket eingeführt
- `\setauthormarkup`, `\setlocextension`, `\setremarkmarkup` für Einstellungen
- Beispieldateien generiert
- LPPL eingefügt

Bugfixes

- Fehler mit *ifthen*-Paketplatzierung behoben
- bei Liste war immer „Eingefügt“ eingestellt, behoben
- Autorausgabe war buggy (`\if`-Abfrage nicht einwandfrei)

6.9 Version 0.1

Datum: 16. 01. 2007

- initiale Version
- Befehle `\added`, `\deleted` und `\changed`

7 Weitergabe, Copyright, Lizenz

Copyright 2007-2010 Ekkart Kleinod (ekkart@ekkart.de)

Dieses Paket darf unter der „*TeX* 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 *TeX*-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

changes.drv
changes.dtx
changes.ins
README

und den generierten Dateien

changes.sty
changes.pdf
changes.example1.tex
changes.example2.tex
changes.example3.tex