

# **pdfcomment.sty**

**v2.4a**

**Ein benutzerfreundliches Interface zu  
PDF-Kommentaren**

**2018/11/01**

Package Autor:  
**Josef Kleber**

## Zusammenfassung

pdflatex bietet seit jeher den Befehl \pdfannot für das Einfügen beliebiger PDF annotations. Allerdings in einer Form, in der zusätzliche Kenntnisse der PDF-Referenz unabdingbar sind. pdfcomment.sty ist eine Antwort auf die in Newsgroups – von Zeit zu Zeit – auftauchende Frage, wie man die Kommentarfunktion des Adobe Reader nutzen könnte. Zumindest aus dem L<sup>A</sup>T<sub>E</sub>X-Code heraus, bietet pdfcomment.sty eine bequeme und nutzerfreundliche Möglichkeit PDF annotations in PDF-Dateien zu nutzen. Seit Version v1.1 unterstützt pdfcomment.sty auch:

L<sup>A</sup>T<sub>E</sub>X → dvips → ps2pdf, L<sup>A</sup>T<sub>E</sub>X → dvipdfmx<sup>1</sup>, XeL<sup>A</sup>T<sub>E</sub>X und L<sup>A</sup>uaL<sup>A</sup>T<sub>E</sub>X.

Leider ist die Unterstützung der PDF annotations durch die PDF-Viewer teilweise nur sehr eingeschränkt bzw. gar nicht vorhanden. Der Referenz-Viewer für die Entwicklung die Pakets ist Adobe Reader.

Falls Sie diesen Kommentar nicht sehen, benutzen Sie definitiv den falschen PDF-Viewer!

## Benötigte Pakete für die Nutzung von pdfcomment.sty

hyperref (v6.76a [2007/04/09]), zref (v1.8 [2007/04/22]), xkeyval, etoolbox, luatex85, ifpdf, iflatax, ifthen, calc, marginnote, datetime2, refcount, soulpos und die von diesen Paketen aufgerufenen Dateien.

---

<sup>1</sup>nur mit der Styleoption dvipdfmx

## Inhaltsverzeichnis

<b>1 Optionen</b>	<b>4</b>
1.1 Globale Styleoptionen . . . . .	4
1.1.1 final . . . . .	4
1.1.2 draft . . . . .	4
1.1.3 dvipdfmx . . . . .	4
1.1.4 version . . . . .	4
1.2 Lokale Optionen . . . . .	4
1.2.1 subject . . . . .	4
1.2.2 author . . . . .	4
1.2.3 opacity . . . . .	5
1.2.4 icon . . . . .	5
1.2.5 deadline . . . . .	5
1.2.6 color . . . . .	5
1.2.7 icolor . . . . .	6
1.2.8 open . . . . .	6
1.2.9 hspace . . . . .	6
1.2.10 type . . . . .	6
1.2.11 font . . . . .	6
1.2.12 fontsize . . . . .	7
1.2.13 fontcolor . . . . .	7
1.2.14 line . . . . .	7
1.2.15 linewidth . . . . .	7
1.2.16 linebegin . . . . .	7
1.2.17 lineend . . . . .	7
1.2.18 linesep . . . . .	8
1.2.19 borderstyle . . . . .	8
1.2.20 dashstyle . . . . .	8
1.2.21 bse . . . . .	8
1.2.22 bsei . . . . .	8
1.2.23 caption . . . . .	8
1.2.24 captionoffset . . . . .	8
1.2.25 captionvoffset . . . . .	8
1.2.26 hoffset . . . . .	8
1.2.27 voffset . . . . .	8
1.2.28 width . . . . .	9
<b>2 Umgebungen</b>	<b>10</b>
2.1 Kommentarumgebungen . . . . .	10
2.1.1 pdfsidelinecomment . . . . .	10
<b>3 Befehle</b>	<b>11</b>
3.1 Kommentarbefehle . . . . .	11
3.1.1 \pdfcomment . . . . .	11
3.1.2 \pdfmargincomment . . . . .	11
3.1.3 \pdfmarkupcomment . . . . .	11
3.1.4 \pdffreetextcomment . . . . .	12
3.1.5 \pdfsquarecomment . . . . .	12
3.1.6 \pdfcirclecomment . . . . .	13
3.1.7 \pdflinecomment . . . . .	13
3.2 Tooltips . . . . .	13
3.2.1 \pdftooltip . . . . .	13
3.3 Allgemeine Befehle . . . . .	14
3.3.1 \pdfcommentsetup . . . . .	14
3.3.2 \listofpdfcomments . . . . .	14
3.3.3 \setliststyle . . . . .	14
3.3.4 \defineliststyle . . . . .	14
3.3.5 \defineavatar . . . . .	15
3.3.6 \definestyle . . . . .	15
<b>4 Drucken der Kommentare und Popups</b>	<b>15</b>

## Danksagung

Für Beiträge zur Entwicklung dieses Pakets möchte ich folgenden Personen danken:

- **Javier Bezos** für die Entwicklung von `soulpos`, das eine bessere Unterstützung von Nicht-Standardtext-Fällen in `\pdfmarkupcomment` möglich macht.
- **Alexander Grahn** für das Bereitstellen eines Patchs für andere Treiber  
 $\text{\LaTeX} \rightarrow \text{dvips} \rightarrow \text{ps2pdf}$ ,  $\text{\LaTeX} \rightarrow \text{dvipdfmx}$  und  $\text{Xe\LaTeX}$ .
- **Ulrike Fischer** für das Beantworten meiner dummen Fragen in d.c.t.t. zur Verwirklichung des Avatar- und Style-Systems und vieles mehr.
- **Christian Feuersänger** für das Einbringen neuer Ideen in Form von `pdfmarginpar.sty` und die Lösung des Druckproblems<sup>2</sup> der PDF annotations und popups.
- **Ross Moore** für das Einbringen von Ideen und Code zur Umsetzung seines eigenen feature requests zur Unterstützung von `\pdfmarkupcomment` im math mode.
- **Heiko Oberdiek** ebenfalls für das Beantworten meiner dummen Fragen in d.c.t.t., sowie die Entwicklung dutzender nützlicher Pakete, speziell `hyperref.sty`, `hycolor.sty` und `zref.sty`, die dieses Paket erst ermöglichten.
- **Herbert Voß** ebenfalls für das Beantworten meiner dummen Fragen in d.c.t.t.

Außerdem möchte ich folgenden Personen danken für bug reports, feature requests, ...: Til Birnstiel, Jannis von Buttlar, Gabriel Cardona, Thomas Feller, Florent Chervet, Jin-Hwan Cho, Marcel Dausend, Andrew Dawson, diabonas, Max Funk, Zvi Gilboa, Thomas König, Marc-André Michel, Guillaume Millet, Fritz Moore, Michael Niedermair, Stefan Pinnow, René Schwarz

---

<sup>2</sup>siehe Abschnitt 4

## 1 Optionen

### 1.1 Globale Styleoptionen

#### 1.1.1 final

Mit der Option `final` versetzt man das Paket in den Final-Modus. Die PDF annotations werden nicht gesetzt und beeinflussen daher auch nicht den Zeilenumbruch. Benutzen Sie die lokale Option `disable`, falls Sie nur einzelne PDF annotations abschalten möchten. (siehe: [1.2.31](#))

#### 1.1.2 draft

Mit der Option `draft` (default) versetzt man das Paket in den Draft-Modus. Die PDF annotations werden daher gesetzt.

#### 1.1.3 dvipdfmx

Falls Sie für die Erstellung Ihrer Dokumente den Treiber `dvipdfmx` einsetzen möchten, müssen Sie die Option `dvipdfmx` benutzen. Die anderen bereits erwähnten Treiber werden automatisch erkannt.

#### 1.1.4 version

Mit der Version v2.0 von `pdfcomment.sty` wurde ein Bugfix hinzugefügt, der ungewollten Whitespace vor den Kommentarbefehlen entfernt. Das ändert u. U. auch den Referenzpunkt für Optionen wie `hoffset`. Möchten Sie das ursprüngliche Verhalten erhalten - etwa für alte Dokumente - verwenden Sie die Option `version=1`. Ansonsten wird die Option `version=2` als Default verwendet!

## 1.2 Lokale Optionen

Die anschließend vorgestellten Optionen sind für die in den Abschnitten [2](#) und [3](#) vorgestellten Befehle anwendbar, allerdings können sie auch als Styleoptionen benutzt werden. Als Styleoptionen besitzen sie globale Wirkung, während sie in Befehlen nur eine lokale Wirkung besitzen und die globalen Optionen nicht überschreiben.

#### 1.2.1 subject

Die Option `subject` dient der Festlegung der Überschriften der PDF popup annotations.

#### 1.2.2 author

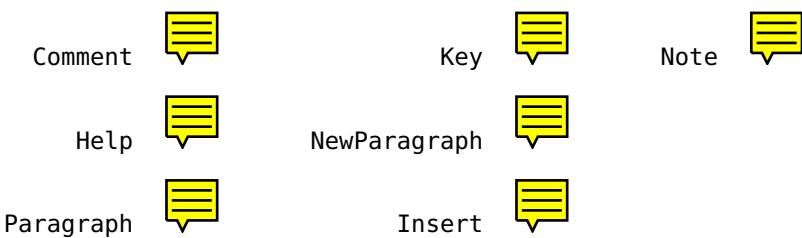
Die Option `author` dient der Festlegung des Verfassers der PDF popup annotations.

### 1.2.3 opacity (1.0)

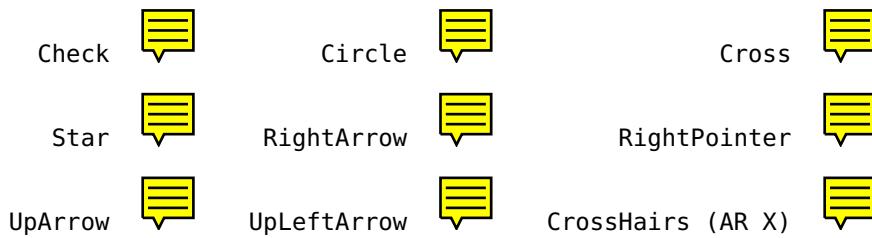
Die Option `opacity` dient der Festlegung der Transparenz der PDF annotations. Die Werte liegen dabei zwischen 0 (transparent) und 1 (nicht transparent). Möchten Sie die PDF popup annotations mit Transparenz drucken, funktioniert das nur zuverlässig, falls sie mit der Option `open=true` erstellt werden.

### 1.2.4 icon (Comment)

Die Option `icon` dient der Festlegung der Grafik, die für die PDF text annotations benutzt werden sollen. Die folgenden Icons werden in der PDF-Referenz als verbindlich festgelegt:



Außerdem werden vom Adobe Reader und einigen anderen Viewern auch diese Icons unterstützt:



### 1.2.5 deadline

Die Option `deadline` dient der Festlegung einer Frist zur Behebung eines Problems. Die Frist wird an das Ende des Kommentars angefügt, sowie gegebenenfalls - bei entsprechendem Listenstil - auch in die Liste der Kommentare aufgenommen.

### 1.2.6 color ([0 0 1] (blau))

Die Option `color` dient der Festlegung der Farbe der PDF annotations z. B. in der Form `{0.34 0.56 0.12}` (RGB). Bei Verwendung des Zusatzpakets `xcolor` kann man auch vordefinierte Basisfarbnamen, sowie die zusätzlich verfügbaren optionalen Farbnamen verwenden. Außerdem kann man mit dem Befehl `\definecolor` eigene benannte Farben definieren. Bitte beachten Sie das beigefügte Beispiel `example.tex`. Es bietet einen Überblick über die unterschiedlichen Möglichkeiten zur Festlegung der Farbe.



Grundsätzlich erlaubt die PDF Spezifikation vier unterschiedliche Farbräume, die sich in der Anzahl der Farbwertangaben unterscheiden:

#	Farbraum
0	transparent
1	Grauwert
3	RGB
4	CMYK

### 1.2.7 **icolor**

Die Option **icolor** dient der Festlegung der bei einigen PDF annotations genutzten sogenannten 'inner color', etwa in Pfeilspitzen oder die Innenfläche der PDF circle annotation. Im Übrigen gelten dieselben Bestimmungen wie bei der Option **color**.

### 1.2.8 **open (true, false)**

Die Option **open** dient der Festlegung des Öffnungsstatus der PDF popup annotations.

Möchten Sie die PDF popup annotations (mit Transparenz) drucken, funktioniert das nur mit PDF annotations, die mit der Option **open = true** erstellt wurden.

### 1.2.9 **hspace (0pt)**

Die Option **hspace** dient der Anpassung des horizontalen Abstands hinter den PDF text annotations, falls der Text nicht durch die PDF text annotations überschrieben werden sollen.

### 1.2.10 **type**

Die Option **type** dient der Festlegung des jeweiligen Typs der PDF annotation, falls ein Kommentarbefehl mehrere PDF annotations unterstützt:

Kommentarbefehl	mögliche Typen
\pdffreetextcomment	freetext, callout, typewriter
\pdflinecomment	line, polyline, polygon

### 1.2.11 **font (Helv)**

Die Option **font** dient der Festlegung der Schriftart in PDF freetext annotations. Möglich sind dabei alle im System installierten Schriften, die keine Leerzeichen im Namen enthalten. Obwohl Adobe Reader diese Schrift als eingebettet kennzeichnet, ist sie das in der Realität nicht und die PDF-Datei enthält nur eine Referenz<sup>3</sup> auf die Schrift (default: Helvetica)! Daher sollte man diese

<sup>3</sup>Adobe Acrobat bietet die Möglichkeit Schriften bei der Konvertierung PS → PDF nachträglich einzubetten

Option nur sehr vorsichtig nutzen und z. B. beim Austausch des Dokuments unter mehreren Autoren nur Schriften nutzen, die auf allen beteiligten Computersystemen vorhanden sind. Das sind etwa die sogenannten 'standard 14 fonts'. 

### 1.2.12 **fontsize (12bp)**

Die Option **fontsize** dient der Festlegung der Schriftgröße in PDF freetext annotations.

### 1.2.13 **fontcolor ([0 0 0] (schwarz))**

Die Option **fontcolor** dient der Festlegung der Schriftfarbe in PDF freetext annotations und erwartet dabei zwingend eine RGB-Farbe.

### 1.2.14 **line**

Die Option **line** dient der Festlegung der Koordinaten der Linien bei bestimmten PDF annotations, z. B. in der Form  $\{x_1 \ y_1 \ x_2 \ y_2\}$ . Der Koordinatenursprung liegt dabei in der linken unteren Seitenecke, wobei die angegebenen Zahlen, als die in PDF-Dokumenten üblichen PostScript-Punkte (L<sup>A</sup>T<sub>E</sub>X-Einheit: bp (big points)) interpretiert werden.

Linientyp	# Punkte
Linie	$n = 2$
Polyline, Polygon	$n > 2$
Callout Linie	$n = 3$

Üblicherweise erfolgt die Angabe der Koordinaten vom Linienanfang zum Linienende. Bei der Callout Linie müssen die Koordinaten vom Linienende bis zum Linienanfang angegeben werden. Beachten Sie dazu bitte das Beispiel [example.pdf](#)!

Zur Vermeidung der Methode 'Versuch und Irrtum' kann man die PDF-Datei mit dem Ghostscript Viewer laden, der in der Lage ist die aktuelle Position des Cursors in der Statuszeile anzuzeigen.

### 1.2.15 **linewidth (1bp)**

Die Option **linewidth** dient der Festlegung der Liniendicke in PDF annotations.

### 1.2.16 **linebegin (/None)**

Die Option **linebegin** dient der Festlegung des Pfeiltyps am Linienanfang. Die Beispieldatei [example.pdf](#) zeigt alle möglichen Pfeiltypen. 

### 1.2.17 **lineend (/OpenArrow)**

Die Option **lineend** dient der Festlegung des Pfeiltyps am Linienende.

## 1.2.18 **linesep (0.5cm)**

Die Option `linesep` dient der Festlegung des horizontalen Abstands der Linie beim Befehl `\pdfsidelinecomment`.

## 1.2.19 **borderstyle (solid,dashed)**

Die Option `borderstyle` dient der Festlegung des Linienstils.

## 1.2.20 **dashstyle ({3 3})**

Die Option `dashstyle` dient der Festlegung des Strichlinienstils, z. B. `{5 3}` (5 Punkte Linie, 3 Punkte Abstand).

## 1.2.21 **bse (none,cloudy)**

Die Option `bse` dient der Festlegung des 'border style effect'.

## 1.2.22 **bsei (1)**

Die Option `bsei` dient der Festlegung der 'border style effect intensity' (Größe der Wolkenelemente). Die PDF-Referenz empfiehlt dabei Werte zwischen 1 und 2.

## 1.2.23 **caption (none,inline,top)**

Die Option `caption` dient der Festlegung des Typs der Linienüberschriften. Die beiden Optionen `inline` und `top` zeigen den Kommentar in der Linie bzw. über der Linie. Bei Verwendung der Option `none` erscheint der Kommentar in einer PDF popup annotation.

## 1.2.24 **captionoffset (0pt)**

Die Option `captionoffset` dient der Festlegung eines horizontalen Offsets, d. h. einer seitlichen Verschiebung, der Linienüberschrift.

## 1.2.25 **captionvoffset (0pt)**

Die Option `captionvoffset` dient der Festlegung eines vertikalen Offsets der Linienüberschrift.

## 1.2.26 **hoffset (0pt)**

Die Option `hoffset` dient der Festlegung eines horizontalen Offsets der PDF annotations.

## 1.2.27 **voffset (0pt)**

Die Option `voffset` dient der Festlegung eines vertikalen Offsets, d. h. einer vertikalen Verschiebung um den angegebenen Wert.

## 1.2.28 width (0pt)

Die Option width dient der Festlegung der Breite der PDF annotations, speziell der Breite der FreeText annotations. PDF text annotations haben grundsätzlich eine Breite von 0pt.

## 1.2.29 height (0pt)

Die Option height dient der Festlegung der Höhe der PDF annotations. Bei PDF text annotations wurde die Höhe auf \normalbaselineskip festgelegt.

## 1.2.30 markup (Highlight,Underline,Squiggly,StrikeOut)

Die Option markup dient der Festlegung des Typs der text markup annotations.

## 1.2.31 disable (true,false)

Mit der Option disable kann man einzelne PDF annotations abschalten. Benutzt man die Option disable global kann man auch mit der Option disable=false lokal einzelne PDF annotations einschalten.

## 1.2.32 date

Mit der Option date kann man den PDF annotations ein Datum zuordnen. Falls kein explizites Datum gesetzt wird, wird das Datum und die Uhrzeit der PDF-Dokument-Erstellung verwendet. Die Eingabe muß im PDF-Datumformat erfolgen, z.B.:

```
D:YYYYMMDDhhmmss+TZ  
D:20101224153657+01'00'
```

## 1.2.33 timezone (+00'00')

Sie müssen die Option timezone nur benutzen, falls Sie die automatische Datumssetzung und Xe<sup>L</sup>T<sub>E</sub>X verwenden. Ansonsten wird die pdflatex-Primitive \pdfcreationdate verwendet. `timezone=+01'00'` setzt die Zeitzone auf mittel-europäische Zeit.

## 1.2.34 avatar

Mit der Option avatar kann man die mit dem Befehl \defineavatar vordefinierten Optionslisten laden, um lästige Tipparbeit zu vermeiden.

## 1.2.35 style

Die Option style dient ebenfalls dem Laden vordefinierter Optionslisten zur Trennung personenbezogener und stylisiter Optionen, z. B. `avatar=Josef, style=MyComment`.

### 1.2.36 **mathstyle** (\textstyle)

Die Option **mathstyle** dient der Korrektur der Größe von PDF annotations - die mit \pdfmarkupcomment oder \pdftooltip im math mode erzeugt wurden - durch Setzen des math style auf \textstyle, \displaystyle, \scriptstyle oder \scriptscriptstyle. Diese Option ist nur lokal für \pdfmarkupcomment und \pdftooltip definiert, daher kann sie nicht global gesetzt werden!

### 1.2.37 **printS0final** (true,false)

Mit der Option **printS0final** kann man die PDF StrikeOut markup annotations inklusive Text automatisch löschen bei gleichzeitiger Nutzung der Option **final** oder **disable**.

## 2 Umgebungen

### 2.1 Kommentarumgebungen

#### 2.1.1 **pdfsidelinecomment**

```
\begin{pdfsidelinecomment}
  [⟨Optionen⟩]⟨Kommentar⟩
\end{pdfsidelinecomment}
```

Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `color`, `icolor`, `opacity`, `linewidth`, `linebegin`, `lineend`, `linesep`, `borderstyle`, `dashstyle`, `caption`, `captionoffset`, `captionvoffset`, `disable`, `date`, `timezone`

Mit der Umgebung **pdfsidelinecomment** kann man ganze Seitenbereiche in Form zweier Linien in den Randspalten mit einem Kommentar kennzeichnen.

Einschränkungen:

- Die PDF sideline annotation darf nicht länger sein als eine Seite, da sonst die Erkennung des Seitenumbruchs fehlschlägt.
- Bei der zwischenzeitlichen Nutzung von dvi-Dateien benötigt man eine L<sup>A</sup>T<sub>E</sub>X-Distribution, die pdf(e)latex in einer Version  $\geq v1.40.0^4$  als Engine benutzt!
- Bei der Verwendung von XeL<sup>A</sup>T<sub>E</sub>X muß Sorge getragen werden, daß die korrekte Seitengröße in die xdvi-Datei geschrieben wird, z. B. über die Option `pagesize` bei Verwendung der KoMa-Script-Klassen, bzw. als Option beim Paket `typearea.sty`.

---

<sup>4</sup>ältere Versionen sind nicht in der Lage im dvi mode x/y-Koordinaten der aktuellen Position zur Verfügung zu stellen.

## 3 Befehle

### 3.1 Kommentarbefehle

Die folgenden Befehlen können Sie zur Kommentierung Ihres Dokuments einsetzen.

#### 3.1.1 \pdfcomment

\pdfcomment  
[*(Optionen)*]<*Kommentar*>

Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `icon`, `color`, `opacity`, `open`, `hspace`, `voffset`, `hoffset`, `disable`, `date`, `timezone`

 \pdfcomment setzt den Kommentar an der aktuellen Position  in den Fließtext.

Das Argument *<Kommentar>* muß dazu intern in PDFDocEncoding/PDFUnicode<sup>5</sup> umgewandelt werden. Einige Zeichen mit spezieller L<sup>A</sup>T<sub>E</sub>X-Bedeutung (&, %, ...) muß man daher escapen oder mit ihrer Befehlsform ersetzen, z. B. \\_ bzw. \textunderscore. Zusätzlich gibt es für die Formatierung die folgenden Befehle:

---

`\textHT` horizontal tab  
`\textLF` line feed  
`\textCR` carriage return

---

#### 3.1.2 \pdfmargincomment

\pdfmargincomment  
[*(Optionen)*]<*Kommentar*>



\pdfmargincomment setzt den Kommentar in Höhe der aktuellen Zeile in die Randspalte. Bitte beachten Sie in diesem Beispiel die Positionierung des Kommentars. Diese Dokumentation wurde mit der L<sup>A</sup>T<sub>E</sub>X-Klasse `ltxdoc` erstellt. Daher erscheint der Kommentar in der linken Randspalte.

#### 3.1.3 \pdfmarkupcomment

\pdfmarkupcomment[*(Optionen)*]  
  *(Markup Text)*<*Kommentar*>

\pdfmarkupcomment setzt eine sogenannte PDF text markup annotation über den mit dem Argument *<Markup Text>* festgelegten Text. Mögliche Typen für die Option `markup` sind dabei: `Highlight`, `Underline`, `Squiggly` und `StrikeOut`

Einschränkungen:

- Wie die Bezeichnung PDF text markup annotation vermuten lässt, eignet sich diese Form der Kommentierung ausschließlich für reinen Text.

---

<sup>5</sup>\hypersetup{unicode}

- Der mit dem Argument  $\langle Markup\ Text \rangle$  festgelegte Text darf nicht länger sein als ein Absatz.
- Die PDF text markup annotation darf nicht länger sein als eine Seite, da sonst die Erkennung des Seitenumbruchs fehlschlägt.
- Bei der zwischenzeitlichen Nutzung von dvi-Dateien benötigt man eine L<sup>A</sup>T<sub>E</sub>X-Distribution, die pdf(e)latex in einer Version  $\geq v1.40.0^6$  als Engine benutzt!
- Bei der Verwendung von XeL<sup>A</sup>T<sub>E</sub>X muß Sorge getragen werden, daß die korrekte Seitengröße in die xdvi-Datei geschrieben wird, z. B. über die Option pagesize bei Verwendung der KoMa-Script-Klassen, bzw. als Option beim Paket typearea.sty.

Seit Version v1.6 funktioniert \pdfmarkupcomment auch im math mode. Sie können jetzt also auch Formeln kommentieren, wie z. B.:

$$\text{Bernoulli Trials: } P(E) = \binom{n}{k} p^k (1-p)^{n-k}$$

Beachten Sie bitte auch [example\\_math\\_markup.pdf](#) für eine genauere Beschreibung der neuen Möglichkeiten!

### 3.1.4 \pdffreetextcomment

`\pdffreetextcomment  
[<Optionen>]<Kommentar>`

Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `color`, `font`, `fontsize`, `fontcolor`, `opacity`, `line`, `linewidth`, `lineend`, `borderstyle`, `dashstyle`, `bse`, `bsei`, `type`, `height`, `width`, `voffset`, `hoffset`, `disable`, `date`, `timezone`

\pdffreetextcomment (type=freetext) setzt einen Kommentar in Form einer frei positionierbaren Box an die gewünschte Stelle der Seite. Bei Verwendung der Option `type=callout` verfügt die Box zusätzlich über eine sogenannte Callout Linie, um die Box an ein bestimmtes Element der Seite zu binden. Mit der Option `type=typewriter` kann man einen Kommentar an einer beliebigen Stelle mit einer virtuellen 'Schreibmaschine' schreiben. Diese PDF freetext annotation haben keinen Rand und einen transparenten Hintergrund. Dadurch kann man beliebige Stellen 'überschreiben'. Bitte beachten Sie dazu die Beispieldatei [example.pdf](#)!

Das ist eine FreeText annotation

### 3.1.5 \pdfsquarecomment

`\pdfsquarecomment  
[<Optionen>]<Kommentar>`

Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `color`, `icolor`, `opacity`, `linewidth`, `borderstyle`, `dashstyle`, `bse`, `bsei`, `height`, `width`, `voffset`, `hoffset`, `disable`, `date`, `timezone`

\pdfsquarecomment setzt einen Kommentar in Form einer rechteckigen Box an die gewünschte Stelle der Seite (siehe: [example.pdf](#)).

---

<sup>6</sup>ältere Versionen sind nicht in der Lage im dvi mode x/y-Koordinaten der aktuellen Position zur Verfügung zu stellen.

### 3.1.6 \pdfcirclecomment

`\pdfcirclecomment[⟨Optionen⟩]{⟨Kommentar⟩}`  
Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `color`, `icolor`, `opacity`, `linewidth`, `borderstyle`, `dashstyle`, `bse`, `bsei`, `height`, `width`, `voffset`, `hoffset`, `disable`, `date`, `timezone`

`\pdfsquarecomment` setzt einen Kommentar in Form eines Kreises oder einer Ellipse an die gewünschte Stelle der Seite (siehe: [example.pdf](#)).

### 3.1.7 \pdflinecomment

`\pdfsidelinecomment[⟨Optionen⟩]{⟨Kommentar⟩}`  
Mögliche Optionen: `avatar`, `style`, `subject`, `author`, `type`, `color`, `icolor`, `opacity`, `caption`, `captionhoffset`, `captionvoffset`, `linewidth`, `linebegin`, `lineend`, `line`, `borderstyle`, `dashstyle`, `disable`, `date`, `timezone`

`\pdflinecomment` setzt einen Kommentar in Form einer Linie, Polylinie oder Polygon an die gewünschte Stelle der Seite (siehe: [example.pdf](#)).

## 3.2 Tooltips

### 3.2.1 \pdftooltip

`\pdftooltip[⟨Optionen⟩]{⟨Objekt⟩}{⟨Kommentar⟩}`  
Mögliche Optionen: `disable`, `mathstyle`

Mit dem Befehl `\pdftooltip` kann man beliebige Objekte mit einem Tooltip versehen - vorausgesetzt man kann das Objekt in eine Box setzen, um die Größe zu messen. Sie können die Tooltips daher problemlos mit **Wörtern** verwenden oder in chemischen Formeln:  $H_2SO_4$

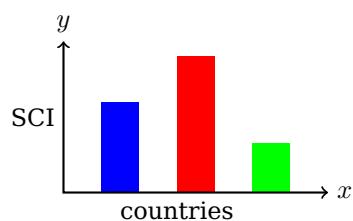
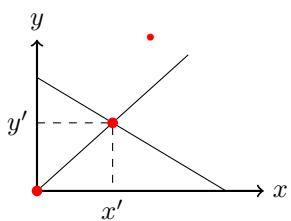
Es funktioniert natürlich auch in abgesetzten Formeln:

$$\sum_{i=1}^n i = \frac{1}{2} n \cdot (n + 1)$$

Bitte beachten Sie auch die Hinweise im Abschnitt [3.1.3](#) zur Verwendung im Mathemodus.

Möchten Sie Teile einer Grafik mit Tooltips versehen bedarf es spezieller Methoden, da `\pdftooltip` in diesen Fällen die Größe nicht selbst bestimmen kann. Bitte beachten Sie dazu die Beispieldatei [example.pdf](#)!

`\pdftooltip` kann nicht durch die Option `final` beeinflusst werden, da die meisten Nutzer die Tooltips sicherlich im finalen Dokument haben möchten. Sie können jedoch mit der Option `disable` ausgeschaltet werden..



## 3.3 Allgemeine Befehle

### 3.3.1 \pdfcommentsetup

\pdfcommentsetup[⟨Optionen⟩]{⟨Optionen⟩}

Mit dem Befehl \pdfcommentsetup kann man jederzeit die globalen Styleoptionen neu setzen. Die optionale Option local hält den Reset lokal (innerhalb einer L<sup>A</sup>T<sub>E</sub>X Gruppe).

### 3.3.2 \listofpdfcomments

\listofpdfcomments[⟨Optionen⟩]

Der Befehl \listofpdfcomments erzeugt eine Liste der Kommentare, um das Auffinden der Kommentare zu vereinfachen. Der Befehl bietet die folgenden Optionen:

- liststyle dient der Festlegung des Listenstils. Dabei sind folgende Werte möglich:
  - ⟨AuthorSubject⟩ (default)
  - ⟨AuthorSubjectDeadline⟩
  - ⟨SubjectAuthorComment⟩
  - ⟨SubjectAuthorCommentDeadline⟩
  - ⟨AuthorComment⟩
  - ⟨AuthorCommentDeadline⟩
  - ⟨SubjectComment⟩
  - ⟨SubjectCommentDeadline⟩
  - ⟨Comment⟩
  - ⟨CommentDeadline⟩
- heading dient der Festlegung der Überschrift, z. B. heading={Liste der Kommentare}.

### 3.3.3 \setliststyle

\setliststyle⟨Name⟩

Der Befehl \setliststyle ist notwendig, falls Kommentare bereits vor \listofpdfcomments eingesetzt werden und ein alternativer Listenstil gewünscht wird.

### 3.3.4 \defineliststyle

\defineliststyle⟨Name⟩{⟨Definition⟩}

Der Befehl \defineliststyle dient der Definition neuer Listenstile, z. B.:

```
\makeatletter
\defineliststyle{MyListstyle}{\pc@lopt@subject\ \textcolor{green}{}
{(\pc@lopt@author)}:\ \pc@lopt@comment\ \textcolor{red}{}
{(\pc@lopt@deadline)}}
\makeatother
```

### 3.3.5 \defineavatar

```
\defineavatar{Name}{Optionen}
```

Der Befehl `\defineavatar` dient der Erstellung benannter vordefinierter Optionslisten, die später in den Kommentarbefehlen über die Option `avatar` wieder geladen werden. Damit können z. B. mehrere Autoren eines Dokuments sehr einfach zwischen ihren Avataren – also ihren grafischen Repräsentationen – wechseln.

### 3.3.6 \definestyle

```
\definestyle{Name}{Optionen}
```

Analog dient `\definestyle` dazu die Optionslisten bei Bedarf in personenbezogene und stylistische Optionslisten aufzuteilen (siehe `example.tex`). Geladen wird die Optionsliste in den Befehlen über die Option `style`.

## 4 Drucken der Kommentare und Popups

Adobe Reader kann prinzipiell PDF annotations drucken. Dazu muß man im Druckdialog im Feld 'Kommentare und Formulare' die Einstellung 'Dokument und Markierungen' wählen. Zusätzlich muß das Drucken im Menü 'Bearbeiten → Voreinstellungen → Kommentieren → Notizen und Popups drucken' konfiguriert werden. Diese Einstellungsmöglichkeit gibt es seit der Veröffentlichung von Adobe Reader X. Bei älteren Versionen bleibt nur die Möglichkeit die Konfigurationsdateien selbst zu ändern.

**Windows:** Ändern der Registry Einträge notwendig (auf eigene Gefahr!)

Starten Sie den Registry Editor (`regedit.exe`) und öffnen Sie z. B. für Adobe Reader 9.0 den folgenden Schlüssel:

`HKEY_CURRENT_USER\Software\Adobe\Acrobat Reader\9.0\Annots\cPrefs`

Der Schlüssel ist für andere Versionen bis auf die Versionsnummer und Groß- und Kleinschreibung ähnlich!

Öffnen oder erstellen Sie (bei geschlossenem Adobe Reader) den Schlüssel `bprintCommentPopups` und ändern Sie den Wert auf 1 (dword).

**Linux:** Öffnen Sie die Konfigurationsdatei:

```
~/.adobe/Acrobat/8.0/Preferences/reader_prefs  
mit einem Text-Editor und ändern Sie den Eintrag  
/printCommentPopups [/b false] zu /printCommentPopups [/b true]
```

Bitte beachten Sie, daß nur geöffnete popup annotations gedruckt werden!