

pdfcomment.sty

v2.2

**Ein benutzerfreundliches Interface zu
PDF-Kommentaren**


23. Mai 2011


Package Autor:
Josef Kleber

Zusammenfassung

pdf_latex 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^AT_EX-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^AT_EX → dvips → ps2pdf, L^AT_EX → dvipdfmx¹ und XeL^AT_EX.

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 der  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, ifpdf, ifthen, calc, marginnote, datetime und die von diesen Paketen aufgerufenen Dateien.

¹nur mit der Styleoption dvipdfmx

Inhaltsverzeichnis

1 Optionen	4		
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 id	4		
1.2.2 subject	4		
1.2.3 author	5		
1.2.4 opacity	5		
1.2.5 icon	5		
1.2.6 deadline	5		
1.2.7 color	5		
1.2.8 icolor	6		
1.2.9 open	6		
1.2.10 hspace	6		
1.2.11 type	6		
1.2.12 font	6		
1.2.13 fontsize	7		
1.2.14 fontcolor	7		
1.2.15 line	7		
1.2.16 linewidth	7		
1.2.17 linebegin	7		
1.2.18 lineend	8		
1.2.19 linesep	8		
1.2.20 borderstyle	8		
1.2.21 dashstyle	8		
1.2.22 bse	8		
1.2.23 bsei	8		
1.2.24 caption	8		
1.2.25 captionhoffset . . .	8		
1.2.26 captionvoffset . . .	8		
1.2.27 hoffset	8		
1.2.28 voffset	9		
1.2.29 width	9		
1.2.30 height	9		
		1.2.31 markup	9
		1.2.32 disable	9
		1.2.33 date	9
		1.2.34 timezone	9
		1.2.35 avatar	9
		1.2.36 style	10
		1.2.37 mathstyle	10
		1.2.38 printSOfinal	10
2 Umgebungen	10		
2.1 Kommentariumgebungen .	10		
2.1.1 pdfsidelinecomment	10		
3 Befehle	11		
3.1 Kommentarbefehle	11		
3.1.1 \pdfcomment	11		
3.1.2 \pdfmargincomment .	11		
3.1.3 \pdfmarkupcomment .	11		
3.1.4 \pdfreetextcomment	12		
3.1.5 \pdfsquarecomment .	12		
3.1.6 \pdfcirclecomment .	13		
3.1.7 \pdflinecomment . .	13		
3.2 Antworten	13		
3.2.1 \pdfreply	13		
3.3 Tooltips	13		
3.3.1 \pdftooltip	13		
3.4 Allgemeine Befehle	14		
3.4.1 \pdfcommentsetup . .	14		
3.4.2 \listofpdfcomments	14		
3.4.3 \setliststyle	14		
3.4.4 \defineliststyle . .	15		
3.4.5 \defineavatar	15		
3.4.6 \definestyle	15		
4 Drucken der Kommentare und Popups	15		
Literatur	17		

Danksagung

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

- **Alexander Grahn** für das Bereitstellen eines Patches für andere Treiber

$\text{\LaTeX} \rightarrow \text{dvips} \rightarrow \text{ps2pdf}$, $\text{\LaTeX} \rightarrow \text{dvi2pdf}$ und XeLaTeX .

- **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 Druckproblems² 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 ermöglichen.
- **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, ...: Jannis von Buttler, Gabriel Cardona, Jin-Hwan Cho, Marcel Dausend, Max Funk, Zvi Gilboa, Thomas König, Marc-André Michel, Guillaume Millet, Fritz Moore, Michael Niedermair, Stefan Pinnow

²siehe 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: [1.2.32](#))

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 id

Die Option `id` dient der Festlegung einer eindeutigen ID für die Referenzierung in einer PDF-Antwort (siehe [3.2.1](#)).

1.2.2 subject

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

1.2.3 author

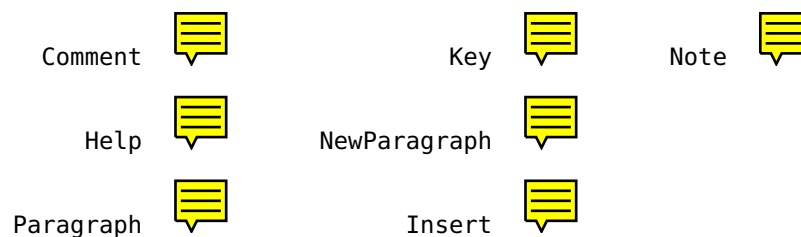
Die Option `author` dient der Festlegung des Verfassers der PDF popup annotations.

1.2.4 opacity (1.0)

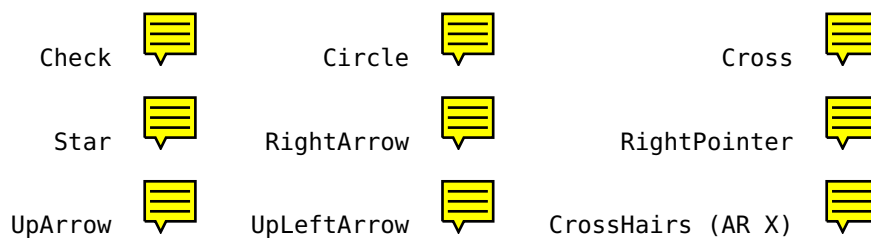
Die Option `opacity` dient der Festlegung der Transparenz der PDF annotations. Die Werte liegen dabei zwischen 0 (transparent) und 1 (nicht transparent). Möchten Sie die PDF popup annotations mit Transparenz drucken, funktioniert das nur zuverlässig, falls sie mit der Option `open=true` erstellt werden.

1.2.5 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:



1.2.6 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.

1.2.7 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

1.2.8 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`.

1.2.9 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.

1.2.10 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.

1.2.11 type

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

Kommentarbefehl	mögliche Typen
<code>\pdfreetextcomment</code>	<u>freetext</u> , callout, typewriter
<code>\pdflinecomment</code>	<u>line</u> , polyline, polygon

1.2.12 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 Referenz³ auf die Schrift (default: Helvetica)! Daher sollte man diese Option nur sehr vorsichtig nutzen und z. B. beim Austausch des Dokuments unter mehreren Autoren nur Schriften nutzen, die auf allen beteiligten Computern vorhanden sind. Das sind etwa die sogenannten 'standard 14 fonts'.

1.2.13 `fontsize` (12bp)

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

1.2.14 `fontcolor` ([0 0 0] (schwarz))

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

1.2.15 `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^AT_EX-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.

1.2.16 `linewidth` (1bp)

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

1.2.17 `linebegin` (/None)

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

³Adobe Acrobat bietet die Möglichkeit Schriften bei der Konvertierung PS → PDF nachträglich einzubetten

1.2.18 lineend (/OpenArrow)

Die Option lineend dient der Festlegung des Pfeiltyps am Linienende.

1.2.19 linesep (0.5cm)

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

1.2.20 borderstyle (solid,dashed)

Die Option borderstyle dient der Festlegung des Linienstils.

1.2.21 dashstyle ({3 3})

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

1.2.22 bse (none,cloudy)

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

1.2.23 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.

1.2.24 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.

1.2.25 captionhoffset (0pt)

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

1.2.26 captionvoffset (0pt)

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

1.2.27 hoffset (0pt)

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

1.2.28 voffset (0pt)

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

1.2.29 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.

1.2.30 height (0pt)

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

1.2.31 markup (Highlight,Underline,Squiggly,StrikeOut)

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

1.2.32 disable (true,false)

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

1.2.33 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'
```

1.2.34 timezone (+00'00')

Sie müssen die Option `timezone` nur benutzen, falls Sie die automatische Datumssetzung und Xe_LA_TE_X oder pdf_latex < v1.30.0 verwenden. Ansonsten wird der pdf_latex-Befehl `\pdfcreationdate` verwendet. `timezone=+01'00'` setzt die Zeitzone auf mitteleuropäische Zeit.

1.2.35 avatar

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

1.2.36 style

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

1.2.37 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!

1.2.38 printSOfinal (true,false)

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

2 Umgebungen

2.1 Kommentariumgebungen

2.1.1 pdfsidelinecomment

<pre>\begin{pdfsidelinecomment} [<Optionen>]{{<Kommentar>}} ... \end{pdfsidelinecomment}</pre>	<p>Mögliche Optionen: <code>avatar</code>, <code>style</code>, <code>subject</code>, <code>author</code>, <code>color</code>, <code>icolor</code>, <code>opacity</code>, <code>linewidth</code>, <code>linebegin</code>, <code>lineend</code>, <code>linesep</code>, <code>borderstyle</code>, <code>dashstyle</code>, <code>caption</code>, <code>captionhoffset</code>, <code>captionvoffset</code>, <code>disable</code>, <code>date</code>, <code>timezone</code></p>
--	---

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^AT_EX-Distribution, die pdf(e)latex in einer Version \geq v1.40.0⁴ als Engine benutzt!
- Bei der Verwendung von XeL^AT_EX 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`.

⁴ältere 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`
`[\langle Optionen \rangle]{\langle Kommentar \rangle}`

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 `{\langle Kommentar \rangle}` muß dazu intern in PDFDocEncoding/PDFUnicode⁵ umgewandelt werden. Einige Zeichen mit spezieller L^AT_EX-Bedeutung (&, %, ...) muß man daher escapen oder mit ihrer Befehlsform ersetzen, z. B. `_` bzw. `\textunderscore` [RO11, 6.2 Definition of `\pdfstringdef`]. Zusätzlich gibt es für die Formatierung die folgenden Befehle:

<code>\textHT</code>	horizontal tab
<code>\textLF</code>	line feed
<code>\textCR</code>	carriage return

3.1.2 `\pdfmargincomment`

`\pdfmargincomment`
`[\langle Optionen \rangle]{\langle Kommentar \rangle}`

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



`\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^AT_EX-Klasse ltxdoc erstellt. Daher erscheint der Kommentar in der linken Randspalte.

3.1.3 `\pdfmarkupcomment`

`\pdfmarkupcomment`
`[\langle Optionen \rangle]{\langle Markup Text \rangle}{\langle Kommentar \rangle}`

Mögliche Optionen: avatar, style, subject, author, color, opacity, markup, mathstyle, disable, date, timezone

`\pdfmarkupcomment` setzt eine sogenannte PDF text markup annotation über den mit dem Argument `{\langle Markup Text \rangle}` 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.

⁵`\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^AT_EX-Distribution, die pdf(e)latex in einer Version $\geq v1.40.0^6$ als Engine benutzt!
- Bei der Verwendung von XeL^AT_EX 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 \pdfreetextcomment

`\pdfreetextcomment`
`[$\langle Optionen \rangle$]{ $\langle Kommentar \rangle$ }`

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

\pdfreetextcomment (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 FreeTex

3.1.5 \pdfsquarecomment

`\pdfsquarecomment`
`[$\langle Optionen \rangle$]{ $\langle Kommentar \rangle$ }`

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](#)).

⁶ältere 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 Antworten

3.2.1 `\pdfreply`

`\pdfreply`
`[<Optionen>]{<Kommentar>}`
 Mögliche Optionen: avatar, style, subject, author, id, replyto, color, disable, date, timezone



Mit dem Befehl `\pdfreply` kann man Antworten zu Kommentaren verfassen. Dabei dient die Option `replyto` der eindeutigen Identifizierung des Kommentars auf den sich die Antwort bezieht⁷. Wegen technischer Gründe funktioniert der Befehl `\pdfreply` ausschließlich bei der Verwendung von `pdflatex`. Weiterhin unterstützt der Befehl keinen Seitenumbruch (`\pdfmarkupcomment`, `\pdfsidelinecomment`). Hier erscheinen die Antworten nur bei den PDF annotations auf der zweiten Seite. Antworten auf Kommentare ohne Seitenumbruch funktionieren problemlos!

3.3 Tooltips

3.3.1 `\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

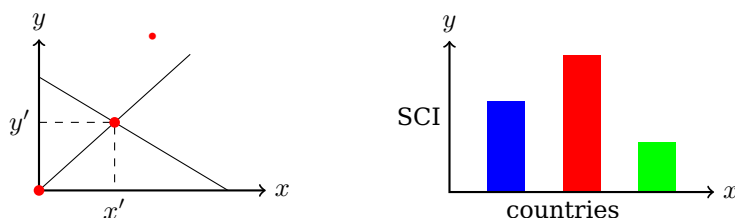
Es funktioniert natürlich auch in abgesetzten Formeln:

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

⁷Die Antworten müssen auf derselben Seite wie die Kommentare gesetzt werden

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](#)!



3.4 Allgemeine Befehle

3.4.1 `\pdfcommentsetup`

`\pdfcommentsetup{<Optionen>}` Mit dem Befehl `\pdfcommentsetup` kann man jederzeit die globalen Styleoptionen neu setzen.

3.4.2 `\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}`.

3.4.3 `\setliststyle`

`\setliststyle{<Name>}` Der Befehl `\setliststyle` ist notwendig, falls Kommentare bereits vor `\listofpdfcomments` eingesetzt werden und ein alternativer Listenstil gewünscht wird.

3.4.4 \defineliststyle

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

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

3.4.5 \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.

3.4.6 \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!

Literatur

- [Ado05] Adobe Systems Incorporated: *pdfmark Reference Manual*. 2005. – URL: http://www.adobe.com/devnet/acrobat/pdfs/pdfmark_reference.pdf
- [Ado06] Adobe Systems Incorporated: *PDF Reference sixth edition*. 2006. – URL: http://www.adobe.com/devnet/acrobat/pdfs/pdf_reference_1-7.pdf
- [Adr08] Adriaens, Hendri: *The xkeyval package*. 2008. – TEXMF://doc/latex/xkeyval/xkeyval.pdf
- [Car99] Carlisle, David: *The file ltxdoc.dtx for use with L^AT_EX*. 1999. – TEXMF://doc/latex/base/ltxdoc.dvi
- [Gün02] Günther, Karsten: *L^AT_EX Ge-packt*. 1. Auflage. mitp, 2002. – ISBN 3-8266-0785-6
- [Ker07] Kern, Uwe: *Extending L^AT_EX 's color facilities: the xcolor package*. 2007. – TEXMF://doc/latex/xcolor/xcolor.pdf
- [Mer02] Merz, Thomas: *Die PostScript & PDF Bibel*. PDFLib edition. dpunkt-Verlag, 2002. – URL: http://www.pdflib.com/fileadmin/pdflib/pdf/Bibel/bibel_d_pdfmark_1x1.pdf
- [MG05] Mittelbach, Frank ; Goossens, Michel: *Der L^AT_EX Begleiter*. 2., überarbeitete und erweiterte Auflage. Pearson Studium, 2005. – ISBN 3-8273-7166-X
- [Oga01] Ogawa, Arthur: *Extensions to the ltxdoc class*. 2001. – TEXMF://doc/latex/revtex4/ltxdocext.pdf
- [RO11] Rahtz, Sebastian ; Oberdiek, Heiko: *Hypertext marks in L^AT_EX*. 2011. – TEXMF://doc/latex/hyperref/hyperref.pdf