

Das pauldoc-Package – Anpassungen für doc für Pauls Package-Dokus*

Paul Ebermann[†]

11. April 2006

Zusammenfassung

Dieses Package enthält einige Befehle, welche die Funktion von `doc` an meine Wünsche anpassen.

Außerdem werden noch einige gewünschte Pakete geladen sowie einige neue Befehle definiert.

Inhaltsverzeichnis

1	Benutzerdoku	2
1.1	Package-Optionen	2
1.2	Neue Makros	2
2	Implementation	3
2.1	Package-Optionen	3
2.2	geladene Pakete	3
2.3	Diverse Einstellungen	3
2.4	Neue oder geänderte Makros	4
2.4.1	Bedingte Anweisungen	4
2.4.2	Packagenamen	4
2.4.3	Logos	4
2.4.4	Lizenz	4
2.4.5	Kopiert aus ltxdoc	6
2.4.6	Bugfixes	7
2.5	Ende	7
A	Liste der Änderungen	8
B	Index	8

*Dieses Dokument gehört zu pauldoc v0.4a, vom 2006/04/05.

[†]Paul-Ebermann@gmx.de

1 Benutzerdoku

1.1 Package-Optionen

chapter Diese Option legt fest, dass Index und Liste der Änderungen in einem eigenen Kapitel (anstatt in einem Abschnitt) beginnen. Dies ist nur in einer Dokumentenklasse sinnvoll, welche überhaupt `\chapter` definiert, etwa den `book`-artigen Klassen.

section Legt fest, dass Index und Änderungsliste in einem Abschnitt beginnen. Dies ist der Vorgabewert.

1.2 Neue Makros

`\ifReferenceExists` `{\refName}{\langle then \rangle}{\langle else \rangle}`

Mit diesem Makro kann man, abhängig davon, ob eine bestimmte Referenz mit `\label{}` gesetzt wurde, bestimmten Text ausgeben.

Falls ein `\ref{\refName}` Erfolg hätte, wird `\langle then \rangle` ausgewertet, ansonsten `\langle else \rangle`. Dies ist nützlich, wenn man im Beschreibungsteil verschiedenen Text aufnehmen will, abhängig davon, ob auch der Implementationsteil mitgesetzt wird – man kann dann etwa auf einen bestimmten Abschnitt darin verweisen, und andernfalls den Text im Konjunktiv formulieren („Wenn der Implementationsteil mit enthalten wäre, könnte man dort jetzt ... finden.“).

`\pack` `{\langle name \rangle}`

In Paketbeschreibungen werden häufiger Paketnamen (der des eigenen Paketes, gelegentlich auch andere Pakete) verwendet. Wie in „How to Package Your L^AT_EX Package“ (Scott Pakin, dtxtut.pdf), Abschnitt 3.2, erläutert, setzt man Paketnamen (sofern es dafür nicht ein eigenes Logo gibt) üblicherweise in `\textsf{}` (die serifenlose Schrift der Klasse). Ich bevorzuge einen sprechenderen Namen, daher gibt es jetzt `\pack{pauldoc}` für `pauldoc`.

`\includeLicense` `\includeLicense` bindet die eventuell im T_EX-Suchpfad zu findende Lizenz ein. Dies ergibt – je nach dem, ob `section` oder `chapter` als Option gegeben wurde – ein neues Kapitel oder einen neuen Abschnitt. Durch Neudefinition von `\lpplfilename` kann festgelegt werden, welche Datei zu verwenden ist – als Vorgabewert ist die bei L^AT_EX mitgelieferte Datei `lppl.tex` im Package festgelegt.

`\lpplfilename`

`\cmd` Diese fünf Makros habe ich aus `ltxdoc` kopiert, damit man sie auch verwenden kann, wenn eine Doku mit einer anderen Klasse formatiert wird (wie bei meiner `algsript`-Klasse). (Sie sind so formatiert, dass eine andere Definition immer vorgeht.)
`\marg`
`\oarg` `\cmd{\langle makro \rangle}` formatiert einen Makronamen. `\cmd{\bla}` ergibt `\bla`. (Sollte der erste Buchstabe kein `\` sein, wird er damit ersetzt: `\cmd{abc}` ergibt `\bc`.)
`\parg` `\cs{\langle makro \rangle}` tut das gleiche – hier wird allerdings der `\` noch angefügt. Falls `\cmd` mal nicht geht, geht gelegentlich noch `\cs`. `\cs{\bla}` ergibt `\bla`.

`\marg{text}` gibt `{\langle text \rangle}` aus. Die Abkürzung steht für „mandatory argument“. `\oarg{text}` ergibt `[\langle text \rangle]` („optional argument“). `\parg{te,xt}` ergibt `(\langle te,xt \rangle)` („picture mode argument“).

2 Implementation

```
1 ⟨*package⟩
```

2.1 Package-Optionen

```
chapter \chapter als Trenner.  
2 \DeclareOption{chapter}{  
3   \let\pauldoc@indexsec=\chapter  
4 }
```

```
section \section als Trenner.  
5 \DeclareOption{section}{  
6   \let\pauldoc@indexsec=\section  
7 }
```

`\pauldoc@indexsec` Wir definieren noch `\pauldoc@indexsec`, rufen dann die Vorgabeoption `section` auf (welche das gleich neu definiert) und verarbeiten die gegebenen Optionen.

```
8 \newcommand*{\pauldoc@indexsec}{}  
9 \ExecuteOptions{section}  
10 \ProcessOptions\relax
```

2.2 geladene Pakete

Wir laden das Paket `inputenc` mit Option `latin1`, um Sonderzeichen auch so eingeben zu können. `babel` sorgt für vernünftige deutsche Bezeichnungen.

```
11 \RequirePackage[latin1]{inputenc}  
12 \RequirePackage[ngerman]{babel}
```

`fontenc` mit Option `T1` lädt das `T1`-Fontencoding, in dem etwa Umlaute direkt aus der Schriftart genommen werden können, anstatt sie mit `¨` und den Vokalen `a`, `o` und `u` zusammenzusetzen. Dies führt auch zu einer verbesserten Silbentrennung.

```
13 \RequirePackage[T1]{fontenc}
```

2.3 Diverse Einstellungen

Wir wollen natürlich in der Doku Querverweise haben, die Änderungen registrieren (für eine entsprechende Liste) und im Index mit Zeilennummern arbeiten. Codestellen wollen wir mit `'` markieren.

```
14 \EnableCrossrefs  
15 \RecordChanges  
16 \CodelineIndex  
17 \AtBeginDocument{\MakeShortVerb{\'}\selectlanguage{ngerman}}
```

Außerdem sollten Index und Änderungsliste auf deutsch beschriftet werden. Wie benutzen hier `\@pauldoc@indexsec`, was entweder `\section` oder `\chapter` ist.

```
18 \renewcommand{\generalname}{Allgemein}  
19 \GlossaryPrologue{\pauldoc@indexsec{Liste der Änderungen}}  
20 \IndexPrologue{\pauldoc@indexsec{Index}}
```

```

21
22   Schrägedruckte Nummern verweisen auf die Seite, auf der der
23   Eintrag beschrieben ist, unterstrichene Nummern zeigen auf die
24   Zeilennummer der Definition, sonstige Zahlen auf die Zeilennummer
25   einer Verwendung.}

```

Der Index bekommt nur zwei Spalten (wir haben einige ziemlich lange Makronamen).

```

26   \setcounter{IndexColumns}{2}

```

2.4 Neue oder geänderte Makros

2.4.1 Bedingte Anweisungen

```

\ifReferenceExists {\label}{\then}{\else}
    \ifReferenceExists prüft einfach, ob das Makro \r@<label> (dort wird die
    Referenz auf das Label gespeichert), existiert, und wertet entsprechend <then>
    oder <else> aus.
27 \newcommand*\ifReferenceExists[3]
28 {%
29   \@ifundefined{r@#1}{#3}{#2}%
30 }

```

2.4.2 Packagenamen

\pack Um Packagenamen wie pauldoc vernünftig auszeichnen zu können, dieser Befehl.

```

31   \newcommand*\pack{\textsf}

```

2.4.3 Logos

\eTeX Aus etex.sty geklaut – das ϵ -TeX-Logo.

```

32 \newcommand*\eTeX{$\m@th\varepsilon$-\TeX}

```

2.4.4 Lizenz

\includeLicense Die Lizenz wird nur eingelesen, wenn sie auch gefunden wird. Dafür wird der Dateiname, welcher in \lplfilename steht, eingebunden.

```

33   \newcommand*\includeLicense {
34     \IfFileExists{\lplfilename}{%

```

Zunächst stellen wir die Sprache um (aber nur Silbentrennung etc., keine Namen), mit dem passenden babel-Befehl.

```

35     \begin{otherlanguage*}{english}

```

Um die Lizenz als \chapter einzubinden und dann Referenzen darauf im Text verwenden zu können, waren mit der L^AT_EX-Version vom 13. Januar 2006 einige Verrenkungen notwendig – siehe unten.

Am 10. Februar habe ich einen Vorschlag zur Verbesserung an die LaTeX-Bug-Liste geschickt, das wurde positiv aufgenommen.¹ Ich habe jetzt von eine

¹Siehe <http://www.latex-project.org/cgi-bin/ltxbugs2html?pr=latex/3839>.

verbesserte Version von Frank Mittelbach bekommen, die hoffentlich demnächst auch veröffentlicht wird.

Wir wollen die Abschnitte der Lizenz auch nummeriert (und im Inhaltsverzeichnis) haben, deswegen die Versionen ohne *. Falls `chapter` als Package-Option angegeben wurde, schieben wir außerdem alles eine Ebene höher.

```

36 \ifx\pauldoc@indexsec\chapter
37   \providecommand*\LPPLsection{\chapter}
38   \providecommand*\LPPLsubsection{\section}
39   \providecommand*\LPPLsubsubsection{\subsection}
40   \providecommand*\LPPLparagraph{\subsubsection}
41 \else
42   \providecommand*\LPPLsection{\section}
43   \providecommand*\LPPLsubsection{\subsection}
44   \providecommand*\LPPLsubsubsection{\subsubsection}
45   \providecommand*\LPPLparagraph{\paragraph}
46 \fi

```

Der folgende (auskommentierte) Code sowie die eingerückte Dokumentation dazu ist mit der neuen Version der Lizenz-Datei nicht mehr notwendig – ich lasse ihn mal trotzdem hier drin, zu Dokumentationszwecken.

Die Lizenz-Datei (zumindest die mir vorliegende Version) verwendet Unterteilungen ab `\section` – falls in unserer Dokumentation `\chapter` verwendet werden soll (d.h. die entsprechende Option gegeben wurde), stellen wir einen Kapitelbeginn davor.

```

\ifx\pauldoc@indexsec\chapter
  \chapter{Lizenz}%
\fi

```

`\section*` *Damit wir aus dem Dokument auf die Lizenz referieren können, definieren wir den Befehl `\section`, welcher zu Beginn der Lizenz verwendet wird, so um, dass er ein Label setzt. (Weil wir uns in einer Gruppe befinden (durch die `otherlanguage`-Umgebung), bleibt die Änderung lokal.)*

```

\newcommand{\lizenz@oldsection@}{}%
\let\lizenz@oldsection@\section\relax%
\def\section*{%
  \label{lppl-chapter}%
  \lizenz@oldsection@*{##1}%
}%

```

`\emph` *Leider habe ich es nicht wirklich geschafft, den Befehl so umzudefinieren, dass nach dem `\section*{}`-Befehl noch ein Label kommt – deswegen definiere ich stattdessen den `\emph{}`-Befehl um, der zumindest in meiner Ausgabe der LPPL direkt nach dem `\section*` kommt. Ein böser Hack, ich weiß ...*

```

\newcommand{\lizenz@oldemph}{}%

```

```

\let\lizenz@oldemph\emph%
\def\emph{%
  \label{lppl-section}%
  \let\emph\lizenz@oldemph%
  \emph%
}%

```

Einige Änderungen sind notwendig, um die Lizenz einzulesen: % soll wieder ein Kommentarzeichen sein, ' darf kein Verbatim-Zeichen mehr sein, weil diese Zeichen in der Lizenz-Datei natürlich nicht den Doc-Konventionen entsprechend verwendet werden. Dann lesen wir die Datei ein, und machen danach die Änderungen wieder rückgängig.

```

47 \DeleteShortVerb{\'}%
48 \MakePercentComment\input{\lpplfilename}\MakePercentIgnore%
49 \MakeShortVerb{\'}%

```

... und jetzt setzen wir die Sprache wieder zurück.

```

50 \end{otherlanguage*}

```

Falls die Lizenz-Datei nicht gefunden wurde, geben wir nur eine passende Meldung aus.

```

51 }%
52 \typeout{^^J%
53 ^^J%
54 Die Datei \lpplfilename{} wurde nicht gefunden.^^J%
55 Schade, da wird die Lizenz eben nicht eingebunden.^^J%
56 ^^J%
57 }%
58 }%
59 }

```

`\lpplfilename` Ich definiere noch den Dateinamen der Lizenz, unter der die meisten meiner Pakete stehen, zur Zeit ist das die unter dem Namen `lppl.tex` verbreitete LPPL.

```

60 \AtBeginDocument{
61 \providecommand{\lpplfilename}{lppl.tex}
62 }

```

2.4.5 Kopiert aus ltxdoc

Die folgenden vier Kommandos kopierte ich aus ltxdoc, um sie auch verwenden zu können, wenn ich nicht diese Klasse verwende. Durch die Verwendung von `\providecommand` gibt es keine Konflikte, falls sie doch schon definiert sind.

`\cmd` Formatiert einen Makronamen, `\cmd{\bla}` ergibt `\bla`. (Sollte der erste Buchstabe kein `\` sein, wird er damit ersetzt.)

```

63 \@ifundefined{cs}
64 {%
65 \providecommand*{\cmd}[1]{\cs{\expandafter\cmd@to\cs\string#1}}
66 \def\cmd@to\cs#1#2{\char\number'#2\relax}

```

`\cs` Implementiert wurde `\cmd` mit `\cs – \cs{bla}` ergibt `\bla`. Dies funktioniert auch an einigen Stellen, wo `\cmd{bla}` nicht funktioniert.²

```

67 \DeclareRobustCommand*\cs[1]{\texttt{\char'\#1}}%
68 }
69 {}

\marg \marg{text} gibt {<text>} aus. Die Abkürzung steht für „mandatory argument“.
70 \providecommand\marg[1]{%
71 {\ttfamily\char'\{} \meta{#1} {\ttfamily\char'\{}}}

\oarg \oarg{text} ergibt [<text>] („optional argument“).
72 \providecommand\oarg[1]{%
73 {\ttfamily[] \meta{#1} {\ttfamily[]}}

\parg \parg{te,xt} ergibt (<te,xt>) („picture mode argument“).
74 \providecommand\parg[1]{%
75 {\ttfamily{} \meta{#1} {\ttfamily{}}}
```

2.4.6 Bugfixes

Inzwischen³ ist der folgende Bug korrigiert, daher hier auskommentiert.

`\SpecialEnvIndex` *Aufgrund eines Bugs in doc⁴ definiere ich hier `\SpecialEnvIndex` neu, damit die environment-Verwendungen genauso wie die environment-Definitionen einsortiert werden, nicht als eigener Eintrag namens „environments:<name>“.*

```

\renewcommand*\SpecialEnvIndex[1]{\@bsphack
\index{#1\actualchar{\protect\ttfamily#1}
(environment)\encapchar usage}%
\index{environments:\levelchar#1\actualchar{%
\protect\ttfamily#1}\encapchar
usage}\@esphack}
```

2.5 Ende

... und Schluss.

```

76 \endinput
77 </package>
```

²Das steht jedenfalls in der ltxdoc-Doku, auch wenn ich das nicht ganz verstanden habe.

³mit dem L^AT_EX-Beta-Release vom 3. Februar 2006 – doc hat das Datum 2006/02/02

⁴Siehe <http://www.latex-project.org/cgi-bin/ltxbugs2html?pr=tools/3834> – das ist (laut einer E-Mail, die ich als Antwort auf meinen Bug-Report (am 2006/01/30) bekam) schon bekannt und im Code korrigiert, ein Release steht bevor.

A Liste der Änderungen

v0.0		\lpplfilename: Dateiname von lppl-1-3b.tex nach lppl.tex geändert. 6
Allgemein: Erste Fassung	1	
v0.1		
Allgemein: \cs, \cmd, \marg, \oarg, \parg hinzugefügt aus ltxdoc. .	6	v0.3
Erste veröffentlichte Fassung ...	1	Allgemein: Jetzt mit \usepackage [T1]{fontenc} 3
\SpecialEnvIndex: Bugfix ist über- flüssig	7	v0.4
v0.2		\eTeX: Neu. 4
\includeLicense: Verwendung ei- ner neuen Version der LPPL macht Workarounds unnötig. .	4	v0.4a
		Allgemein: Kleinere Änderungen an der Dokumentation. 1

B Index

Schräggedruckte Nummern verweisen auf die Seite, auf der der Eintrag beschrieben ist, unterstrichene Nummern zeigen auf die Zeilennummer der Definition, sonstige Zahlen auf die Zeilennummer einer Verwendung.

Symbols		E	
\'	17, 47, 49	\else	41
\ifundefined	29, 63	\emph	<u>47</u>
\\	67	\EnableCrossrefs	14
\{	71	\end	50
\}	71	\endinput	76
		\eTeX	<u>32</u>
A		\ExecuteOptions	9
\AtBeginDocument	17, 60	\expandafter	65
		F	
B		\fi	46
\begin	35		
		G	
C		\generalname	18
\chapter	<u>2</u> , 2, 3, 36, 37	\GlossaryPrologue	19
\char	66, 67, 71		
\cmd	2, <u>63</u>	I	
\cmd@to@cs	65, 66	\IfFileExists	34
\CodelineIndex	16	\ifReferenceExists	2, <u>27</u>
\cs	2, 65, <u>67</u>	\ifx	36
		\includeLicense	2, <u>33</u>
D		\IndexPrologue	20
\DeclareOption	2, 5	\input	48
\DeclareRobustCommand	67		
\def	66	L	
\DeleteShortVerb	47	\let	3, 6

<code>\lizenz@oldsection@</code>	<u>47</u>	<code>\providecommand</code>	
<code>\lpplfilename</code>	2, 34, 48, 54, <u>60</u>	..	37–40, 42–45, 61, 65, 70, 72, 74
<code>\LPPLparagraph</code>	40, 45		
<code>\LPPLsection</code>	37, 42		R
<code>\LPPLsubsection</code>	38, 43	<code>\RecordChanges</code>	15
<code>\LPPLsubsubsection</code>	39, 44	<code>\relax</code>	10, 66
		<code>\renewcommand</code>	18
		<code>\RequirePackage</code>	11–13
	M		
<code>\m@th</code>	32		S
<code>\MakePercentComment</code>	48	<code>\section</code>	2, <u>5</u> , 6, 38, 42
<code>\MakePercentIgnore</code>	48	<code>\section*</code>	<u>47</u>
<code>\MakeShortVerb</code>	17, 49	<code>\selectlanguage</code>	17
<code>\marg</code>	2, <u>70</u>	<code>\setcounter</code>	26
<code>\meta</code>	71, 73, 75	<code>\SpecialEnvIndex</code>	<u>76</u>
	N	<code>\string</code>	65
<code>\newcommand</code>	8, 27, 31–33	<code>\subsection</code>	39, 43
<code>\number</code>	66	<code>\subsubsection</code>	40, 44
	O		T
<code>\oarg</code>	2, <u>72</u>	<code>\TeX</code>	32
	P	<code>\textsf</code>	31
<code>\pack</code>	2, <u>31</u>	<code>\texttt</code>	67
<code>\paragraph</code>	45	<code>\ttfamily</code>	71, 73, 75
<code>\parg</code>	2, <u>74</u>	<code>\typeout</code>	52
<code>\pauldoc@indexsec</code> ..	3, 6, <u>8</u> , 19, 20, 36		V
<code>\ProcessOptions</code>	10	<code>\varepsilon</code>	32