Automatisch “Tagged PDF” mit LaTeX erzeugen
Was ist heutzutage möglich?

Ulrike Fischer, Bonn
LaTeX Project Team

6.8.2023

FrOSCon

Was ist LaTeX

- Textsatzsystem und Markup-Sprache
- älter als HTML/PDF/Unicode/CSS: erste Version um 1984 von Leslie Lamport
- weitverbreitet im MINT Bereich
- Overleaf hat mehr als 10 Millionen Nutzer
- 2^+ Millionen Documente in `arXiv.org`

LaTeX: Source example

```
\documentclass{demo1} % or demo2
\author{Frank Mittelbach}
\title{Example \LaTeX{} document}
\begin{document}

\maketitle

\tableofcontents

\section{Introduction}

This example shows\footnote{See
  \url{https://latex-project.org} for more.}
\begin{itemize}
  \item the title
  \item the table of contents
  \item section headings
  \item a list
  \item some text
  \item a footnote
  \item some math
  \item a figure.
\end{itemize}

\subsection{Some Math}
A famous equation
\begin{equation} E= mc^2 \end{equation}

\section{Sample text}
Take a look at figure~\vref{fig:cups}.

\kant[1][1]

\begin{figure}\centering
\includegraphics[width=\linewidth]{coffeecup}
\caption{Two coffee cups\label{fig:cups}}
\end{figure}

\kant[2] \kant[3][1-4] \kant[4]
\end{document}
```

LaTeX: Formatted output I

Example LaTeX document

Frank Mittelbach

September 6, 2022

Contents

1 Introduction

This example shows[1]

- the title
- the table of contents
- section headings
- a list
- some text
- a footnote
- some math
- a figure.

[1] See `https://latex-project.org` for more.

Figure 1: Two coffee cups

1.1 Some Math

A famous equation

$$E = mc^2 \tag{1}$$

2 Sample text

Take a look at figure 1.

As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves; as I have shown elsewhere, the phenomena should only be used as a canon for our understanding.

Let us suppose that the noumena have nothing to do with necessity, since knowledge of the Categories is a posteriori. Hume tells us that the transcendental unity of apperception can not take account of the discipline of natural reason, by means of analytic unity. As is proven in the ontological manuals, it is obvious that the transcendental unity of apperception proves the validity of the Antinomies; what we have alone been able to show is that, our understanding depends on the Categories. It remains a mystery why the Ideal stands

LaTeX: Formatted output II

Example LaTeX document

Frank Mittelbach

September 6, 2022

Contents

Introduction

This example shows[1]

- the title
- the table of contents
- section headings
- a list
- some text
- a footnote
- some math
- a figure.

[1] See https://latex-project.org for more.

Some Math

A famous equation

(1) $E = mc^2$

Sample text

Take a look at figure 1 on the following page.

As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves; as I have shown elsewhere, the phenomena should only be used as a canon for our understanding.

Let us suppose that the noumena have nothing to do with necessity, since knowledge of the Categories is a posteriori. Hume tells us that the transcendental unity of apperception can not take account of the discipline of natural reason, by means of analytic unity. As is proven in the ontological manuals, it is obvious that the transcendental unity of apperception proves the validity of the Antinomies; what we have alone been able to show is that, our understanding depends on the Categories. It remains a mystery why the Ideal stands in need of reason. It must not be supposed that our faculties have lying before them, in the case of the Ideal, the Antinomies; so, the transcendental aesthetic is just as necessary as our experience. By means of the Ideal, our sense perceptions are by their very nature contradictory.

As is shown in the writings of Aristotle, the things

Figure 1: Two coffee cups

in themselves (and it remains a mystery why this is the case) are a representation of time. Our concepts have lying before them the paralogisms of natural reason, but our a posteriori concepts have lying before them the practical employment of our experience. Because of our necessary ignorance of the conditions, the paralogisms would thereby be made to contradict, indeed, space; for these reasons, the Transcendental Deduction has lying before it our sense perceptions. (Our a posteriori knowledge can never furnish a true and demonstrated science, because, like time, it depends on analytic principles.

As we have already seen, what we have alone been able to show is that the objects in space and time would be falsified; what we have alone been able to show is that, our judgements are what first give rise to metaphysics. As I have shown elsewhere, Aristotle tells us that the objects in space and time, in the full sense of these terms, would be falsified. Let us sup-

LaTeX: Beispiele

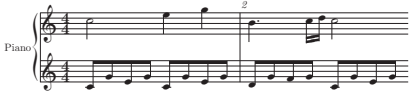
Everything written symbols can say has already passed by ...

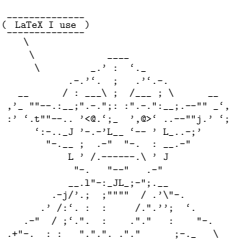
THEY ARE LIKE TRACKS LEFT BY ANIMALS. That is why the masters of meditation refuse to accept that writings are final. The aim is to reach true being by means of those tracks, those letters, those signs; but reality itself is not a sign, and it leaves no tracks. It doesn't come to us by way of letters or words. We can go toward it, by following those words and letters back to whence they came from.

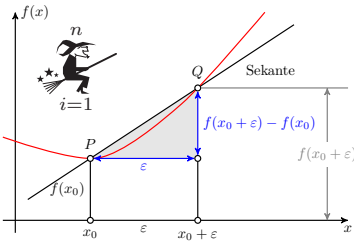
إِيَّاكَ نَعْبُدُ وَإِيَّاكَ نَسْتَعِينُ

μὴ θορυβεῖτε, ὦ ἄνδρες Ἀθηναῖοι

সবকিছু প্রবাহিত হয়

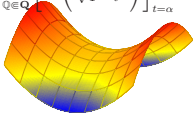




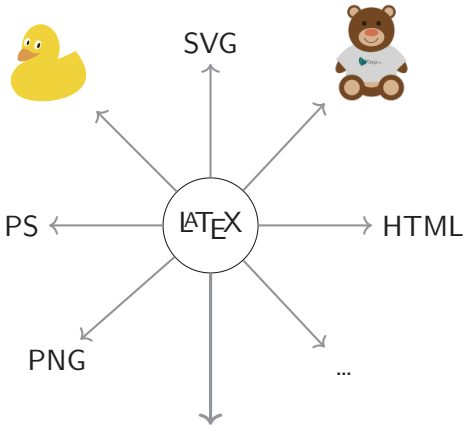


$$\iiint_{Q} f(w,x,y,z)\,dw\,dx\,dy\,dz \le \oint_{\partial Q} f'\left(\max\left\{\frac{\|w\|}{|w^2+x^2|};\frac{\|z\|}{|y^2+z^2|};\frac{\|w\oplus z\|}{\|x\oplus y\|}\right\}\right)$$

$$\biguplus_{\mathbb{Q}\in\bar{\mathbb{Q}}} \left[f^*\left(\frac{\mathbb{Q}(t)}{\sqrt{1-t^2}}\right)\right]_{t=\alpha}^{t=\vartheta} \gtrsim -(\Delta+\nu-v)^3$$



LaTeX kann unterschiedliche Formate erzeugen

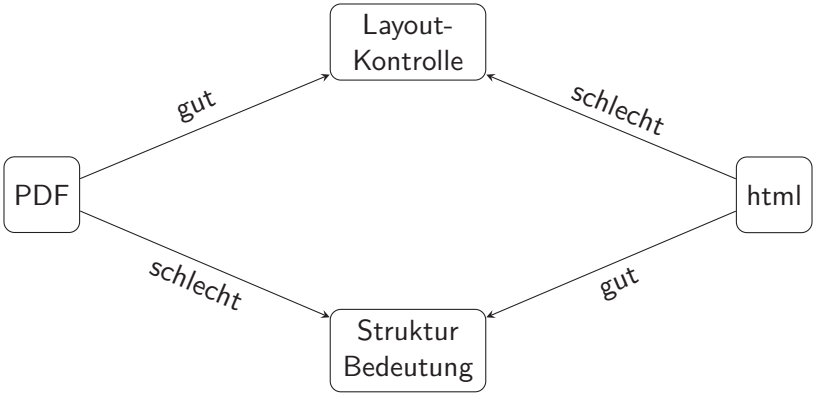


Aber PDF ist derzeit das meist verwendete

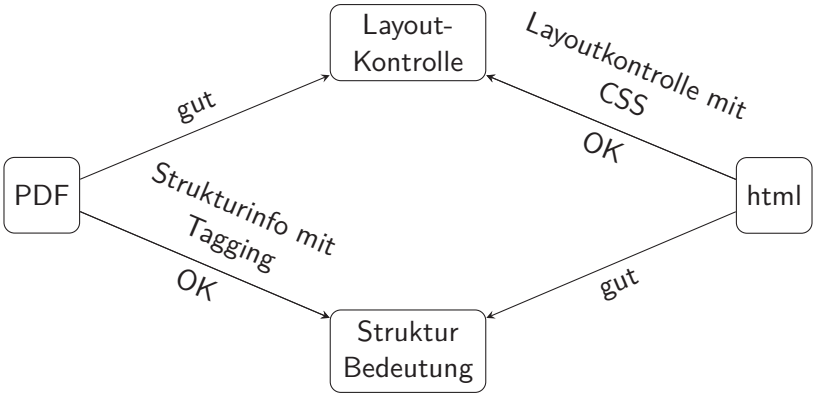
Viele Wege führen zu PDF ...

- LaTeX + `pdftex` = pdfLaTeX
- LaTeX + `luatex` = luaLaTeX
- LaTeX + `xetex` = xeLaTeX
- LaTeX + `pdftex` (dvimode) + `dvipdfmx`
- dvilualaTeX + `dvipdfmx`
- dvilualaTeX + `xdvipsk` (extern) + `ps2pdf`
- LaTeX + `dvips` + `ps2pdf`
- LaTeX + `dvips` + `distiller`

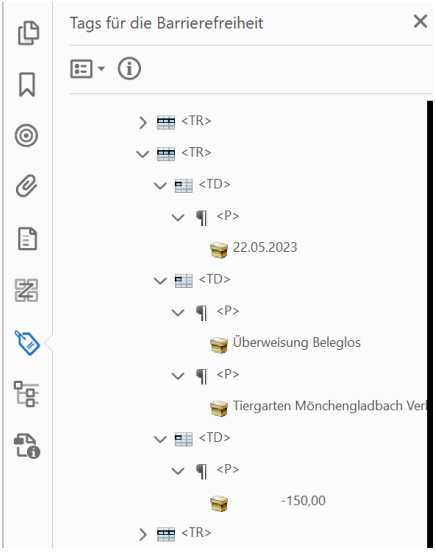
Warum Tagging? PDF ist ein seiten- und graphikorientiertes Format



Warum Tagging? PDF ist ein seiten- und graphikorientiertes Format



Motivation: Ein getaggter Kontoauszug



Ein getaggter Kontoauszug

- Simples Tagging
- ein Tabelle mit drei Spalten: Datum, mehrzeilige Beschreibung, Betrag

Motivation: Ein getaggter Kontoauszug – Vorteile

- Vorlesen
  - nicht getagged: Reihenfolge ist durcheinander
  - getagged: Reihenfolge ist korrekt, Spalten- und Zeilennummern sind bekannt
- Copy & Paste in eine Tabellenkalkulation
  - nicht getagged: alles wird in eine Zelle gesteckt
  - getagged: Daten werden korrekt in drei Spalten gesplittet

Also warum Tagging?

- Strukturinformationen verbessern Barrierefreiheit
- Strukturinformationen verbessern die Wiederverwendung von Daten
- Strukturinformationen verbessern Reflow

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream

BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ




/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ




169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream

BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ




/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ




169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream

BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ




/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ




169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream

BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ




/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ




169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream
/H1 <</MCID 0>> BDC
BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ
ET
EMC

BT
/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ
ET


BT
169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream
/H1 <</MCID 0>> BDC
BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ
ET
EMC
/P <</MCID 1>> BDC
BT
/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ
ET
EMC

BT
169.365 -593.872 Td [(1)]TJ
ET

endstream
```

Wie geht Tagging? – Schritt 1: Marked Content (MC-chunks)

```
stream
/H1 <</MCID 0>> BDC
BT
/F17 14.3462 Tf 124.802 706.129 Td [(1)-1100(Section)]TJ
ET
EMC
/P <</MCID 1>> BDC
BT
/F15 9.9626 Tf 0 -21.819 Td [(hallo)]TJ
ET
EMC
/Artifact <</Type /Pagination>> BDC
BT
169.365 -593.872 Td [(1)]TJ
ET
EMC
endstream
```

Wie geht Tagging? – Schritt 2: Strukturobjekte (StructElem)

```
1 0 obj
<< /Type /StructTreeRoot ....
   /K 4 0 R >> endobj
```

/K = Kid
/P = Parent
/S = Subtype

Wie geht Tagging? – Schritt 2: Strukturobjekte (StructElem)

```
1 0 obj
<< /Type /StructTreeRoot ....
   /K 4 0 R >> endobj

4 0 obj
<< /Type /StructElem /S /Document
   /P 1 0 R /K [5 0 R 7 0 R]>> endobj
```

/K = Kid
/P = Parent
/S = Subtype

Wie geht Tagging? – Schritt 2: Strukturobjekte (StructElem)

```
1 0 obj
<< /Type /StructTreeRoot ....
   /K 4 0 R >> endobj

4 0 obj
<< /Type /StructElem /S /Document
   /P 1 0 R /K [5 0 R 7 0 R]>> endobj

5 0 obj
<< /Type /StructElem /S /H1
   /P 4 0 R /K <</Type /MCR /Pg 6 0 R /MCID 0>>>> endobj
```

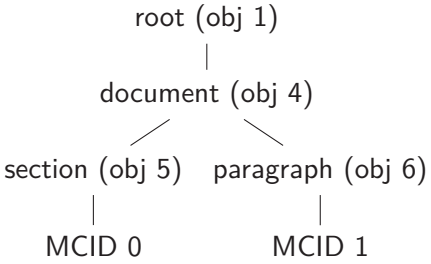
/K = Kid
/P = Parent
/S = Subtype

Wie geht Tagging? – Schritt 2: Strukturobjekte (StructElem)

```
1 0 obj
<< /Type /StructTreeRoot ....
   /K 4 0 R >> endobj

4 0 obj
<< /Type /StructElem /S /Document
   /P 1 0 R /K [5 0 R 7 0 R]>> endobj

5 0 obj
<< /Type /StructElem /S /H1
   /P 4 0 R /K <</Type /MCR /Pg 6 0 R /MCID 0>>>> endobj

7 0 obj
<< /Type /StructElem /S /P
   /P 4 0 R /K <</Type /MCR /Pg 6 0 R /MCID 1>>>> endobj
```

/K = Kid
/P = Parent
/S = Subtype

Wie geht Tagging? – Strukturbaum



Wie geht Tagging? – Schritt 3: Bürokratie

Mehr Objekte und sonstige Dinge ...

- Dokumentation `tagpdf` Paket
- Diverse tugboat Artikel

Manuelles Tagging in LaTeX– ein einfaches Beispiel

```
\DocumentMetadata{testphase=phase-I}%lädt und aktiviert tagpdf
```

Manuelles Tagging in LaTeX– ein einfaches Beispiel

```
\DocumentMetadata{testphase=phase-I}%lädt und aktiviert tagpdf
\documentclass{article}
\begin{document}
```

Manuelles Tagging in LaTeX– ein einfaches Beispiel

```
\DocumentMetadata{testphase=phase-I}%lädt und aktiviert tagpdf
\documentclass{article}
\begin{document}

\tagstructbegin{tag=H1}% Struktur H1
 \tagmcbegin{}% MC
 \section{A section}
 \tagmcend% Ende MC
\tagstructend% Ende Struktur
```

Manuelles Tagging in LaTeX– ein einfaches Beispiel

```
\DocumentMetadata{testphase=phase-I}%lädt und aktiviert tagpdf
\documentclass{article}
\begin{document}

\tagstructbegin{tag=H1}% Struktur H1
 \tagmcbegin{}% MC
 \section{A section}
 \tagmcend% Ende MC
\tagstructend% Ende Struktur

\tagstructbegin{tag=P}% Struktur P
 \tagmcbegin{}% MC
 Ein Absatz ...
 \tagmcend% Ende MC
\tagstructend% Ende Struktur

\end{document}
```

Ziele des Tagged PDF Projektst

- Tagging mit LaTeX *ermöglichen* $\Rightarrow$ done!
- Tagging mit LaTeX soll *automatisch* und *einfach* sein

Problem: Tags sind unsichtbar

- Freie PDF Viewer zeigen Tags nicht
  (einige Ausnahmen: PDF-XChange, PDFix Desktop Lite)
- Was man nicht sieht, das nutzt man nicht
- Testen der Struktur ist schwierig für Anwender

Problem: Fehlender PDF 2.0 support

PDF 2.0 ist wichtig für gutes Tagging

- Mehr und sinnvollere Tags: Title, Aside, FEnote ...
- "structure destinations" für Links zu Strukturen
- "associated files" um Daten an Strukturen zu koppeln
- MathML Namensraum

} wichtig für Math Tagging

LaTeX kann "tagged PDF 2.0" aber

PDF Viewer unterstützen PDF 2.0 Features nicht

Problem: Parent-Child-Regeln

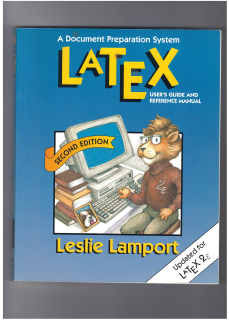
- PDF-Strukturen müssen Parent-Child-Regeln befolgen:

Structure Type	Children		Parents	
	Occ.	Structure Type	Occ.	Structure Type
P	0..n	NonStruct	0..n	Document
	0..n	Private	0..n	DocumentFragment
	0..n	Note	‡	Part
	0..n	Code	‡	Div
	0..n	Sub	0..n	Art
	0..n	Lbl	0..n	Sect
	0..n	Em	0..n	TOCI
	0..n	Strong	0..n	Aside

- Die Regeln passen nicht immer LaTeX Strukturen
- Die Einhaltung der Regeln muss geprüft und Fehler müssen korrigiert werden

Nächster Meilenstein: Tagging von “Leslie Lamport Dokumenten”

- Standardklassen
- Standardbefehle und -umgebungen
- begrenzter Satz an Paketen
- hyperref

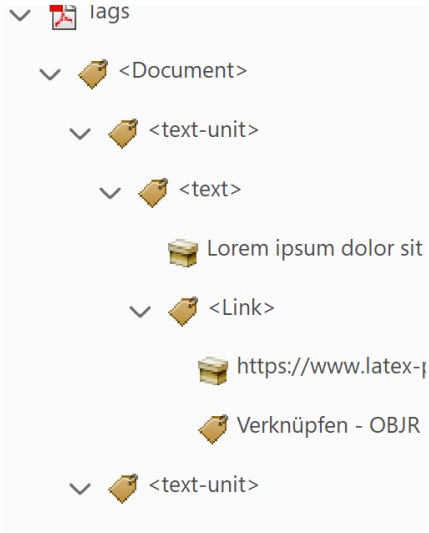


Was ist heutzutage möglich?

- today = LaTeX 2023-06-01
- Engines: pdfLaTeX or luaLaTeX
- Code im `latex-lab` Bundle
- Code wird mit `testphase` Keys geladen:

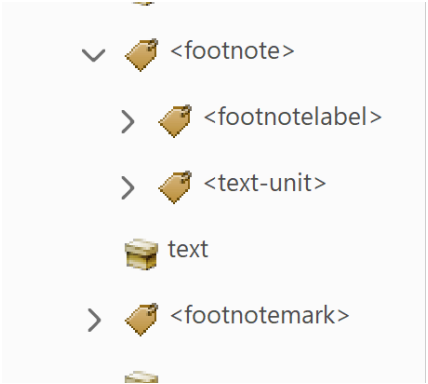
```
\DocumentMetadata{testphase=phase-III}
\documentclass{article}
```

Absätze und Links



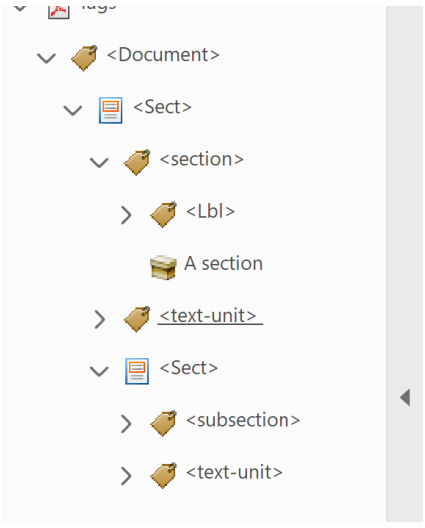
- implementiert mit den neuen para Hooks

Fußnoten



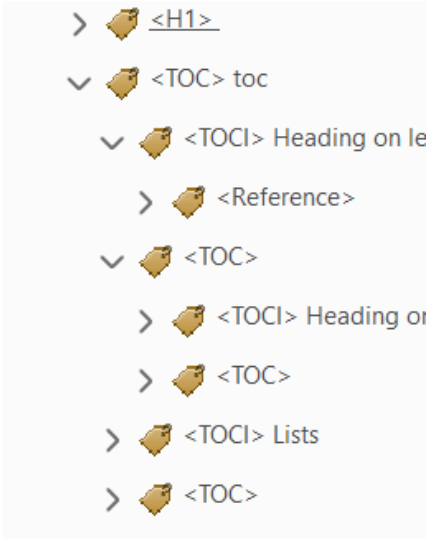
- neue Implementation
- `footmisc` ist bereits kompatibel
- Fußnoten in `minipage` im nächstem LaTeX-dev

Überschriften



- Sect-Struktur umgibt Titel und Text
- ändert:
  `\@startsection`, `\@sect` etc
- derzeit incompatibel:
  e.g. `memoir`, KOMA-Klassen, `titlesec`

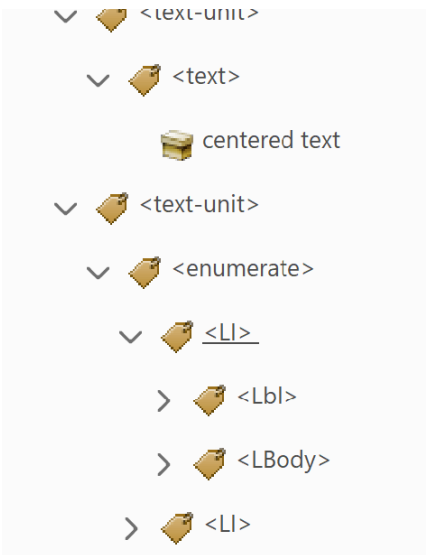
Table of contents and ähnliche Listen



- geändert:
  `\@starttoc`, `\addcontentsline`, `\@dottedtocline` and `\l@chapter` etc

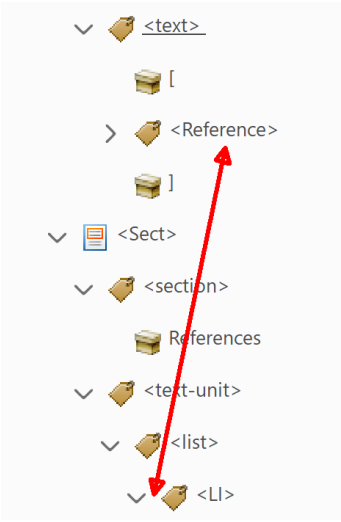
- derzeit inkompatibel:
  z.B. `memoir`, KOMA-Klassen, `titletoc`

Display-Umgebungen and Listen



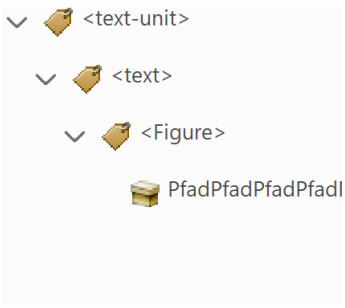
- LaTeX-Umgebungen die auf `trivlist` basieren:
  - `center`, `flushleft`, `flushright`
  - `quotation`, `quote`, `verse`
  - `verbatim`
  - `theorems`
  - `enumerate`, `itemize`, `description`
  - `list`, `trivlist`
- Völlig neue Implementation mit `xtemplate`
- Block-Umgebungen wie `center` oder `verbatim` sind keine Listen mehr
- derzeit incompatibel:
  `enumitem`, `enumerate`, `fancyvrb`, `listings`, theorem-Pakete ...

Quellenverweise und Bibliographie



- `thebibliography` wird getaggt
- `natbib` kann genutzt werden
- `biblatex` mit `hyperref` auch
- fehlt noch:
  `biblatex` ohne `hyperref`

Bilder



- Keys für alternativen Text:

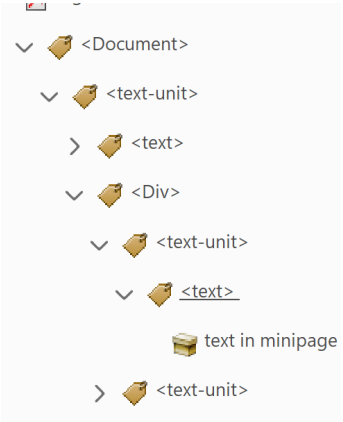
```
\includegraphics[alt={This shows a
    duck}]{duck}

\begin{picture}[alt={This shows a duck
    too}](100,100)
...
\end{picture}
```

- `tikz` wird noch nicht unterstützt

Gleitumgebungen (Floats)

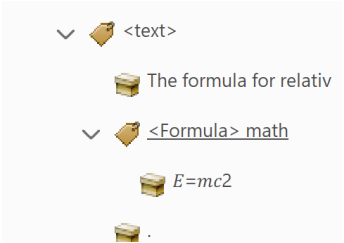
- geändert:
  `\@xfloat` uns `\@makecaption`
- `caption` Paket ist mehr oder weniger kompatibel
- `float` Paket ist derzeit inkompatibel
- `\marginpar` wird noch nicht unterstützt.

minipage und \parbox



- Tagging ist aktiviert
- Anpassungen in Spezialfällen nötig

Mathematik



- experimentaler Prototyp für math tagging
- wird als additionales Modul geladen:

```
\DocumentMetadata
 {testphase={phase-III,math}}
```

- Inhalt wird zuerst als ganzes gelesen und gespeichert und dann weiterverarbeitet.
- kann auch “non-math” Text wie urls erfassen
- nur minimales Tagging, optimales Tagging unklar (fehlender PDF 2.0 support)

“Firstaid” Support für Pakete

- `firstaid` Key lädt kleine Korrekturen für Klassen und Pakete

```
\DocumentMetadata{testphase={phase-III,firstaid}}
```

Was fehlt? Wiederverwenden von Boxen

- `\savebox` und `\usebox` werden noch nicht unterstützt.

Was fehlt? Tabellen

- Manuelles Tagging ist möglich
- Wie identifiziert man *automatisch* “header”-Zellen?

	Studio	Apt	Chalet	Villa
Paris				
1 bedroom	11	20	25	23
2 bedroom	-	43	52	32
3 bedroom	-	13	15	40
Rome				
1 bedroom	13	21	22	3
2 bedroom	-	23	43	30
3 bedroom	-	16	32	40

Quelle: https://www.w3.org/WAI/tutorials/tables

Zusammenfassung

- Automatisches Tagging ist nun für diverse Standarddokumente aktiviert:

```
\DocumentMetadata {testphase={phase-III,math,firstaid}}
\documentclass{book}
\usepackage[math,toc]{blindtext}
\begin{document}
\Blinddocument
\end{document}
```

- Tester und Feedback sind willkommen!
- Bugreports, Fragen und Diskussionen:
  `https://github.com/latex3/tagging-project`

Thank you for your attention!