

Wissenschaftl. Textverarbeitung mit \LaTeX

WS 2015/16 - 5. Vorlesung

Alexander Richter

Institut für Mathematische Optimierung

30. November 2015

In der vorletzten Woche wurde u.a. behandelt:

- ▶ Tabellen mit `booktabs.sty`
- ▶ Mathematischer Formelsatz mit \LaTeX

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

Nachtrag: Spaltenbezogene Befehle (Präambel)

>{ }c der Inhalt der geschweiften Klammern wird an den Anfang der c-Zelle gesetzt und dort verarbeitet

c<{ } der Inhalt der geschweiften Klammern wird an das Ende der c-Zelle gesetzt und dort verarbeitet

@{ } der Inhalt der geschweiften Klammern wird zwischen Zwei Spalten gesetzt und ersetzt den Standardspaltenabstand

- ▶ beiden oberen Befehle benötigen array-Paket
- ▶ array-Paket wird automatisch von booktabs geladen

Nachtrag: Spaltenbezogene Befehle (Präambel)

- >{ }c der Inhalt der geschweiften Klammern wird an den Anfang der c-Zelle gesetzt und dort verarbeitet
 - c<{ } der Inhalt der geschweiften Klammern wird an das Ende der c-Zelle gesetzt und dort verarbeitet
 - @{ } der Inhalt der geschweiften Klammern wird zwischen Zwei Spalten gesetzt und ersetzt den Standardspaltenabstand
- ▶ **beiden oberen Befehle benötigen array-Paket**
 - ▶ array-Paket wird automatisch von booktabs geladen

Nachtrag: Spaltenbezogene Befehle (Präambel)

- >{ }c der Inhalt der geschweiften Klammern wird an den Anfang der c-Zelle gesetzt und dort verarbeitet
 - c<{ } der Inhalt der geschweiften Klammern wird an das Ende der c-Zelle gesetzt und dort verarbeitet
 - @{ } der Inhalt der geschweiften Klammern wird zwischen Zwei Spalten gesetzt und ersetzt den Standardspaltenabstand
- ▶ **beiden oberen Befehle benötigen array-Paket**
 - ▶ **array-Paket wird automatisch von booktabs geladen**

- \LaTeX : Tabellen sind boxen \rightarrow optionale “box-parameter”

a				a
b				b
c				c
d				d
[c] (default) Grundlinie	a	[t] Grundlinie	d	[b] Grundlinie
	b			
	c			
	d			

```
\begin{tabular}[c]{c}a\\b\\c\\d\end{tabular}
\begin{tabular}[t]{c}a\\b\\c\\d\end{tabular}
\begin{tabular}[b]{c}a\\b\\c\\d\end{tabular}
```

Tabellenumgebungen

- ▶ `tabular`
- ▶ `tabular*`
- ▶ `tabularx`

Spaltentypen

- ▶ `l`, `c`, `r`
- ▶ `p{ }`
- ▶ `X`

Abstandsanpassungen

- ▶ `\tabcolsep`
- ▶ `\arraystretch`
- ▶ `\extracolsep{\fill }`

externe Ausrichtung

- ▶ `c`
- ▶ `t`
- ▶ `b`

```
\begin{tabularx}{\linewidth}{|X|X|X|}  
Text ... & Text ... & Text ... \\  
\end{tabularx}
```

Die Forelle (*Salmo trutta*) ist die in Europa wohl bekannteste Vertreterin der Forellenfische. Sie ist mit dem atlantischen Lachs (*Salmo salar*) gattungsverwandt und tritt in mehreren Unterarten auf.

Die Forelle ist ein hervorragender Speisefisch. Forelle Mül-
lerin (in Mehl gewendet und
gebraten, kann gut mit dün-
nen, geschälten Zitronenschei-
ben und Tomatenscheiben an-
gerichtet werden) ist die klassi-
sche Art der Zubereitung. Wei-
terhin bietet es sich an, die Fo-
relle zu braten oder als Forelle
blau mit Essig zu köcheln. Sehr
gut schmeckt die Forelle auch
geräuchert.

Bananen (*Musa*) sind eine Gat-
tung in der Familie der Ba-
nanengewächse (*Musaceae*) in-
nerhalb der einkeimblättrigen
Pflanzen. In der Gattung gibt
es rund 100 Arten.

Noch einmal tabularx.sty

```
\begin{tabularx}{\linewidth}{@{}l
>{\setlength\hspace{.5\hspace}}X
>{\setlength\hspace{1.5\hspace}}X@{}}
Text ... & Text ... & Text ... \\
\end{tabularx}
```

Forellen Bananen (Musa) sind eine Gattung in der Familie der Bananengewächse (Musaceae) innerhalb der einkeimblättrigen Pflanzen. In der Gattung gibt es rund 100 Arten.

Die Forelle ist ein hervorragender Speisefisch. Forelle Müllerin (in Mehl gewendet und gebraten, kann gut mit dünnen, geschälten Zitronenscheiben und Tomatenscheiben angerichtet werden) ist die klassische Art der Zubereitung. Weiterhin bietet es sich an, die Forelle zu braten oder als Forelle blau mit Essig zu köcheln. Sehr gut schmeckt die Forelle auch geräuchert. Bananen (Musa) sind eine Gattung in der Familie der Bananengewächse (Musaceae) innerhalb der einkeimblättrigen Pflanzen. In der Gattung gibt es rund 100 Arten.

- Trick: berechnetes `\hspace`-Attribut der 'X'-Spalte nutzen

```
\begin{tabularx}{\linewidth}{@{}l
>{\setlength\hspace{.5\hspace}}X
>{\setlength\hspace{1.5\hspace}}X@{}}
Text ... & Text ... & Text ... \\
\end{tabularx}
```

Forellen Bananen (Musa) sind eine Gattung in der Familie der Bananengewächse (Musaceae) innerhalb der einkeimblättrigen Pflanzen. In der Gattung gibt es rund 100 Arten.

Die Forelle ist ein hervorragender Speisefisch. Forelle Müllerin (in Mehl gewendet und gebraten, kann gut mit dünnen, geschälten Zitronenscheiben und Tomatenscheiben angerichtet werden) ist die klassische Art der Zubereitung. Weiterhin bietet es sich an, die Forelle zu braten oder als Forelle blau mit Essig zu köcheln. Sehr gut schmeckt die Forelle auch geräuchert. Bananen (Musa) sind eine Gattung in der Familie der Bananengewächse (Musaceae) innerhalb der einkeimblättrigen Pflanzen. In der Gattung gibt es rund 100 Arten.

- Summe der Vorfaktoren == # variabler Spalten ('X'-Spalten)

Neue Spalten definieren: `\newcolumntype`

`\newcolumntype{<Name>}{<Definition>}`

- ▶ Mehrfachverwendung von Spaltendefinitionen → Effizienz
- ▶ übersichtlicherer Quellcode

Neue Spalten definieren: \newcolumnntype

```
\newcolumnntype{R}{>{\raggedright\arraybackslash}X}  
\begin{tabularx}{\linewidth}  
{R|R|R}  
Text ... & Text ... & Text ... \\  
\end{tabularx}
```

Die Forelle (*Salmo trutta*) ist die in Europa wohl bekannteste Vertreterin der Forellenfische. Sie ist mit dem atlantischen Lachs (*Salmo salar*) gattungsverwandt und tritt in mehreren Unterarten auf.

Die Forelle ist ein hervorragender Speisefisch. Forelle Müllerin (in Mehl gewendet und gebraten, kann gut mit dünnen, geschälten Zitronenscheiben und Tomatenscheiben angerichtet werden) ist die klassische Art der Zubereitung. Weiterhin bietet es sich an, die Forelle zu braten oder als Forelle blau mit Essig zu köcheln. Sehr gut schmeckt die Forelle auch geräuchert.

Bananen (*Musa*) sind eine Gattung in der Familie der Bananengewächse (*Musaceae*) innerhalb der einkeimblättrigen Pflanzen. In der Gattung gibt es rund 100 Arten.

Neue Spalten mit Parametern: \newcolumnntype

```
\newcolumnntype{<Name>}[n]{<Definition>}  
\newcolumnntype{L}[1]{>{\RaggedRight}p{#1}}  
\begin{tabular}{\linewidth}  
{L{3cm}|L{5cm}}  
Text ... & Text ... & Text ... \\  
\end{tabular}
```

Die Forelle (*Salmo trutta*) ist die in Europa wohl bekannteste Vertreterin der Forellenfische. Sie ist mit dem atlantischen Lachs (*Salmo salar*) gattungsverwandt und tritt in mehreren Unterarten auf.

Die Forelle ist ein hervorragender Speisefisch. Forelle Müllerin (in Mehl gewendet und gebraten, kann gut mit dünnen, geschälten Zitronenscheiben und Tomatenscheiben angerichtet werden) ist die klassische Art der Zubereitung. Weiterhin bietet es sich an, die Forelle zu braten oder als Forelle blau mit Essig zu köcheln. Sehr gut schmeckt die Forelle auch geräuchert.

Problem

- ▶ c-Spalten unbrauchbar
- ▶ r-Spalten unbrauchbar

1.2	1.2	1.2	1,2
1.23	1.23	12.5	300,2
1121.2	1121.2	861.20	674,29
184	184	10	69
.4	.4		,4

Problem

- ▶ c-Spalten unbrauchbar
- ▶ r-Spalten unbrauchbar

1.2	1.2	1.2	1,2
1.23	1.23	12.5	300,2
1121.2	1121.2	861.20	674,29
184	184	10	69
.4	.4		,4

Neuer Spaltentyp D

`D{Eingabezeichen}{Ausgabezeichen}{Dezimalstellenzahl}`

Eingabezeichen:	Separatorzeichen in der TEX-Datei
Ausgabezeichen:	Separatorzeichen in der Ausgabedatei
Dezimalstellenzahl:	maximale Zahl der Dezimalstellen

Achtung: D-Spalten werden im Mathematikmodus gesetzt, also in den hierfür voreingestellten Schriften.

```
\begin{tabular}{rD{,}{.}{2}}  
\toprule  
200,67 & 200,67\\  
10,1 & 10,1 \\  
\bottomrule  
\end{tabular}
```

200,67	200.67
10,1	10.1

```
\begin{tabular}{rD{,}{-}{2}}  
\toprule  
200,67 & 200,67\\  
10,1 & 10,1 \\  
\bottomrule  
\end{tabular}
```

200,67	200–67
10,1	10–1

Dezimalspalten: dcolumn.sty

```
\begin{tabular}  
{|D{.}{,}{-2}|D{.}{,}{2}|D{.}{.}{2}|D{,}{.}{2}|}  
1.2    & 1.2    & 1.2    & 1,2 \\  
1.23   & 1.23   & 12.5    & 300,2 \\  
1121.2& 1121.2& 861.20  & 674,29 \\  
184    & 184    & 10      & 69 \\  
.4     & .4     &         & ,4 \\  
\end{tabular}
```

1,2	1,2	1.2	1.2
1,23	1,23	12.5	300.2
1121,2	1121,2	861.20	674.29
184	184	10	69
,4	,4		.4

Dezimalspalten: dcolumn.sty

```
\begin{tabular}
{|D{.}{,}{-1}|D{.}{,}{5}|D{.}{.}{2}|D{,}{.}{2}|}
1.2      & 1.2    & 1.2     & 1,2 \\
1.23     & 1.23   & 12.5    & 300,2 \\
1121.200& 1121.2& 861.20  & 674,29 \\
184      & 184    & 10      & 69 \\
.4       & .4     &         & ,4 \\
\end{tabular}
```

1,2	1,2	1.2	1.2
1,23	1,23	12.5	300.2
1121,200	1121,2	861.20	674.29
184	184	10	69
,4	,4		.4

Minus vor Zahl der Dezimalstelle → Die Spalte wird so dimensioniert, dass das Separatorzeichen in der Mitte steht.

- ▶ `longtable.sty`
Gleiche Spaltenbreite über alle Seiten hinweg (kann mehrere Durchläufe erfordern, Spaltenbreiteninformationen werden in die `.aux`-Datei geschrieben); nicht verwendbar für mehrspaltigen Satz. Kein Seitenumbruch innerhalb von Zellen.
- ▶ `supertabular.sty` Tabelle wird in kleinere Tabellen aufgebrochen. Kein Seitenumbruch innerhalb von Zellen
- ▶ In diesem Kurs: nur `longtable.sty`

- ▶ `longtable.sty`
Gleiche Spaltenbreite über alle Seiten hinweg (kann mehrere Durchläufe erfordern, Spaltenbreiteninformationen werden in die `.aux`-Datei geschrieben); nicht verwendbar für mehrspaltigen Satz. Kein Seitenumbruch innerhalb von Zellen.
- ▶ `supertabular.sty` Tabelle wird in kleinere Tabellen aufgebrochen. Kein Seitenumbruch innerhalb von Zellen
- ▶ In diesem Kurs: nur `longtable.sty`

```
\begin{longtable}[ausrichtung]{spaltendeklarationen}  
...  
\end{longtable}
```

(Horizontale) Ausrichtungsmöglichkeiten (außen): c,l,r oder über Längen \LTleft und \LTRight.
(default: center)

Vertikale Abstände zum vorangehenden/nachfolgenden Material:
\LTpre, \LTpost (default: \bikskipamount)

Beispiel!

Tabellen: longtable.sty

Das vorangehende Bsp. wurde erzeugt mit (in etwa):

```
\begin{longtable}{lll ...}  
T-Nr.&Blatt 1& Blatt 2 & ... & Gesamt \\  
\midrule  
\endfirsthead  
T-Nr.&Blatt 1& Blatt 2 & ... & Gesamt \\  
\midrule  
\endhead  
\bottomrule  
\multicolumn{9}{c}{(Fortsetzung auf der nächsten Seite)}  
\endfoot  
% nichts %  
\bottomrule  
\endlastfoot  
001 &5&10&10&11&11&&&47\\  
002 &5&9&10&12&13&&&49\\  
003 &5&10&9&12&&&&36 ...  
\end{longtable}
```

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

Zähler sind Variablen, die einen ganzzahligen Wert besitzen können und der Nummerierung aller möglichen Dinge dienen.

Einige der \LaTeX -eigenen Zähler sind z.B.

Zähler	Abfrage
page	<code>\thepage</code>
section	<code>\thesection</code>
subsection	<code>\thesubsection</code>
equation	<code>\theequation</code>
figure	<code>\thefigure</code>
footnote	<code>\thefootnote</code>
enumi	<code>\theenumi</code>

Beispiel `\subsection`:

2.3 Ein Unterabschnitt

Abgesehen von der Ausgabe des übergebenen Textes als Überschrift, passiert Folgendes:

- ▶ Erhöhung des Zählers `subsection` um den Wert Eins
- ▶ Ausgabe der beiden Zähler `section` und `subsection` mittels Punkt getrennt
- ▶ Rücksetzung des Zählers `subsubsection` auf Null

Für die Ausgabe stehen folgende Befehle bereit:

Befehl	Ergebnis
<code>\arabic{page}</code>	1
<code>\Roman{page}</code>	I
<code>\roman{page}</code>	i
<code>\Alph{page}</code>	A
<code>\alph{page}</code>	a

Die the-Befehle nutzen meist `\arabic`, können aber (mittels `\renewcommand`) umdefiniert werden und so auch weitere Ausgabebefehle enthalten.

Für die Ausgabe stehen folgende Befehle bereit:

Befehl	Ergebnis
<code>\arabic{page}</code>	2
<code>\Roman{page}</code>	II
<code>\roman{page}</code>	ii
<code>\Alph{page}</code>	B
<code>\alph{page}</code>	b

Die the-Befehle nutzen meist `\arabic`, können aber (mittels `\renewcommand`) umdefiniert werden und so auch weitere Ausgabebefehle enthalten.

Section–Seite¹

2–3; Oder: 2–3. Oder: II–iii. Oder: B–c.

`\thesection--\thepage;`

Oder: `\arabic{section}--\arabic{page}.`

Oder: `\Roman{section}--\roman{page}.`

Oder: `\Alph{section}--\alph{page}.`

¹Der Unterschied zwischen Inhalt des Seiten-Zählers und der Nummer unten rechts auf den Folien liegt daran, dass einige Folien aus mehr als einer Seite bestehen aufgrund des schrittweisen Aufdeckens.

Ebenso können eigene Zähler definiert werden:
(sollte im Vorspann/Header erfolgen!)

```
\newcounter{Name}[Rücksetzer]
```

Achtung: Kein \ wie bei Befehlen im Namen!

Rücksetzer ist der Name eines anderen Zählers, der den neu definierten Zähler auf Null setzen kann (vgl. subsection vs. dessen Rücksetzer section).

Ebenso können eigene Zähler definiert werden:
(sollte im Vorspann/Header erfolgen!)

```
\newcounter{Name}[Rücksetzer]
```

Achtung: Kein \ wie bei Befehlen im Namen!

Rücksetzer ist der Name eines anderen Zählers, der den neu definierten Zähler auf Null setzen kann (vgl. subsection vs. dessen Rücksetzer section).

Mittels `\setcounter{Name}{Wert}` kann jederzeit ein Wert gesetzt werden.

Mittels `\addtocounter{Name}{Wert}` kann jederzeit der Wert verändert werden. (Auch Verminderung möglich!)

Mittels `\setcounter{Name}{Wert}` kann jederzeit ein Wert gesetzt werden.

Mittels `\addtocounter{Name}{Wert}` kann jederzeit der Wert verändert werden. (Auch Verminderung möglich!)

`\stepcounter{Name}`

`\refstepcounter{Name}`

Zähler Name wird um 1 erhöht, alle Zähler, die Name als Rücksetzer haben, werden auf 0 gesetzt.

`\refstepcounter` ermöglicht zusätzlich, mittels `\label` und `\ref` Verweise auf den Zähler zu konstruieren.

Sinnvoll für Einsatz in Befehlen und Umgebungen!

`\stepcounter{Name}`

`\refstepcounter{Name}`

Zähler Name wird um 1 erhöht, alle Zähler, die Name als Rücksetzer haben, werden auf 0 gesetzt.

`\refstepcounter` ermöglicht zusätzlich, mittels `\label` und `\ref` Verweise auf den Zähler zu konstruieren.

Sinnvoll für Einsatz in Befehlen und Umgebungen!

`\value{Name}`

liefert den Zählerwert als Zahlwert zur \LaTeX -internen Weiterverwendung, z.B. für andere Zähler.

Ausgabe:

mittels `\theName` (funktioniert auch bei eig. Zählern!) oder den schon vorgestellten Ausgabeformatierungen `\arabic`, `\Roman`, ...

`\value{Name}`

liefert den Zählerwert als Zahlwert zur \LaTeX -internen Weiterverwendung, z.B. für andere Zähler.

Ausgabe:

mittels `\theName` (funktioniert auch bei eig. Zählern!) oder den schon vorgestellten Ausgabeformatierungen `\arabic`, `\Roman`, ...

Zur Seitenzahl soll 3 dazugezählt werden und das Ergebnis in kleinen römischen Ziffern ausgegeben werden.

```
\newcounter{Seitenno}
```

```
\setcounter{Seitenno}{\value{page}}
```

```
\addtocounter{Seitenno}{3}
```

Auf der Seite `\thepage` erhalten wir `\roman{Seitenno}`.

Auf der Seite 40 erhalten wir xliii.

Sie möchten eine `enumerate`-Umgebung mit großen und kleinen römischen Zahlen in der ersten bzw. zweiten Gliederungsstufe:

- I. Außen 1.
 - I.i. Innen 1.1.
 - I.ii. Innen 1.2.
- II. Außen 2.
 - II.i. Innen 2.1.
 - II.ii. Innen 2.2.
- III. Außen 3.

Umsetzung mit Zählern:

```
\newcounter{aussen}
\newcounter{innen}[aussen]
\newcommand{\INIT}{
  \setcounter{aussen}{0}\setcounter{innen}{0}}
\newcommand{\itema}{
  \stepcounter{aussen}
  \item[\Roman{aussen}. ]}
\newcommand{\itemb}{
  \stepcounter{innen}
  \item[\Roman{aussen}.\roman{innen}.]}
```

Zähler: Beispiel (Forts.)

```
\begin{enumerate}\INIT
  \itema Außen 1.
  \begin{enumerate}
    \itemb Innen 1.1.
    \itemb Innen 1.2.
  \end{enumerate}
  \itema Außen 2.
  \begin{enumerate}
    \itemb Innen 2.1.
    \itemb Innen 2.2.
  \end{enumerate}
  \itema Außen 3.
\end{enumerate}
```

- I. Außen 1.
 - I.i. Innen 1.1.
 - I.ii. Innen 1.2.
- II. Außen 2.
 - II.i. Innen 2.1.
 - II.ii. Innen 2.2.
- III. Außen 3.

Beim Übergang von I. zu II. sorgt der Befehl `\stepcounter` dafür, dass der Zähler innen (der ja den Rücksetzer aussen hat) auf 0 zurückgesetzt wird.

- 1 Tabellen III
- 2 Genauer: eigene Zähler
- 3 Genauer: Eigene Umgebungen

Definition von Umgebungen

```
\newenvironment{Name}[anz][std]{Anfang}{Ende}  
\renewenvironment{Name}[anz][std]{Anfang}{Ende}
```

Name Frei wählbarer Name der Umgebung

anz Anzahl der übergebenen Argumente

std Standard-Belegung für erstes Argument

Dieses ist dann optional. (funktioniert genauso wie bei Befehlen!)

Anfang Durch was soll „\begin{Name}“ ersetzt werden?

Ende Durch was soll „\end{Name}“ ersetzt werden?

Hier kein Zugriff auf die Argumente mehr!

Definition von Umgebungen

`\newenvironment{Name}[anz][std]{Anfang}{Ende}`

`\renewenvironment{Name}[anz][std]{Anfang}{Ende}`

Name Frei wählbarer Name der Umgebung

anz Anzahl der übergebenen Argumente

std Standard-Belegung für erstes Argument

Dieses ist dann optional. (funktioniert genauso wie bei Befehlen!)

Anfang Durch was soll „`\begin{Name}`“ ersetzt werden?

Ende Durch was soll „`\end{Name}`“ ersetzt werden?

Hier kein Zugriff auf die Argumente mehr!

Beispiel

```
\newenvironment{Zitat}  
{  
  \noindent\textbf{Zitat: }  
  \begin{quote}  
  }{  
  \end{quote}  
}
```

Anwendungsbeispiel

```
\begin{Zitat}
```

```
Dies ist ein Zitat!
```

```
\end{Zitat}
```

oder

```
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}
```

Ausgabe

Zitat:

Dies ist ein Zitat!

Anwendungsbeispiel

```
\begin{Zitat}
```

```
Dies ist ein Zitat!
```

```
\end{Zitat}
```

oder

```
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}
```

Ausgabe

Zitat:

Dies ist ein Zitat!

Leider klappt das noch nicht so wie gewünscht!

Eingabe

```
Vorangehende Zeile\ldots  
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}  
Nachfolgende Zeile\ldots
```

Ausgabe

```
Vorangehende Zeile... Zitat:  
    Dies ist ein Zitat!  
Nachfolgende Zeile...
```

Leider klappt das noch nicht so wie gewünscht!

Eingabe

```
Vorangehende Zeile\ldots  
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}  
Nachfolgende Zeile\ldots
```

Ausgabe

Vorangehende Zeile... **Zitat:**

Dies ist ein Zitat!

Nachfolgende Zeile...

Abhilfe durch par (neuer Absatz, wie Leerzeile):

Verbesserte Umgebung

```
\newenvironment{Zitat}  
{\par\noindent\textbf{Zitat: }  
  \begin{quote}}  
\end{quote}}
```

Jetzt ist das Problem gelöst:

Eingabe

```
Vorangehende Zeile\ldots  
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}  
Nachfolgende Zeile\ldots
```

Ausgabe

```
Vorangehende Zeile...  
Zitat:  
    Dies ist ein Zitat!  
Nachfolgende Zeile...
```

Jetzt ist das Problem gelöst:

Eingabe

```
Vorangehende Zeile\ldots  
\begin{Zitat} Dies ist ein Zitat! \end{Zitat}  
Nachfolgende Zeile\ldots
```

Ausgabe

Vorangehende Zeile...

Zitat:

Dies ist ein Zitat!

Nachfolgende Zeile...

Beispiel

```
\newenvironment{Zitat}[2][Douglas Adams]
{\par\noindent\textbf{Zitat }( #1)
  \begin{quote}\itshape}
{\end{quote}
 \small \mbox{} \hfill ( #2) \par}}
```

Fehlermeldung! Im Teil Ende dürfen keine Argumente übergeben werden!

Abhilfe mit „gespeicherten Boxen“

<code>\newsavebox{\Name}</code>	Deklariert neue Box „\Name“
<code>\sbox{\Name}{TEXT}</code>	Speichert „TEXT“ in der Box „\Name“
<code>\usebox{\Name}</code>	Verwendet die gespeicherte Box

Eine einfache Anwendung

Eingabe

```
\newsavebox{\Test}  
\sbox{\Test}{TEXT}  
Viel weiter unten\newline  
\usebox{\Test}
```

Ausgabe

```
Viel weiter unten  
TEXT
```

Saveboxen sind keine *Quellcode-Ersetzer* wie eigene Befehle, sondern funktionieren wie eine wiederverwendbare `makebox`.

Beachten Sie dies insbesondere bei Formatierungen:

Beim Schreiben in die Box ist alles möglich, beim Verwenden der Box sind Formatierungen innerhalb der Box nicht mehr möglich.

Probieren Sie es aus!

(vgl. auch Wirkung des `\small` in den nachfolgenden Beispielen)

Eigene Umgebungen mit Argumenten

Damit können wir auch am Ende der Umgebung Zitat ein Argument verarbeiten

Zitate mit Quellenangaben

```
\newsavebox{\Quelle}  
\newenvironment{Zitat}[2][Douglas Adams]  
{\sbox{\Quelle}{#2}  
  \par\noindent\textbf{Zitat }( #1)  
  \begin{quote}\itshape  
\end{quote}  
  {\small \mbox{}\hfill (Quelle: \usebox{\Quelle})\par}}
```


Eingabe

```
\begin{Zitat}{Das Restaurant am Ende des Universums}  
... \end{Zitat}
```

Ausgabe

Zitat (Douglas Adams)

Am Anfang wurde das Universum erschaffen. Das machte viele Leute sehr wütend und wurde allenthalben als Schritt in die falsche Richtung angesehen.

(Quelle: Das Restaurant am Ende des Universums)

Eingabe

```
\begin{Zitat}[Tony Stark]{Filmzitat}  
Ich bin Iron Man.  
\end{Zitat}
```

Ausgabe

Zitat (Tony Stark)
Ich bin Iron Man.

(Quelle: Filmzitat)

Natürlich kann man diesen Zitaten auch zu einer Nummer verhelfen.

Zitate mit fortlaufender Nummer

```
\newcounter{ZitatNr}  
\newenvironment{Zitat}[2][Douglas Adams]  
{\sbox{\Quelle}{#2}\refstepcounter{ZitatNr}  
  \par\noindent  
  \textbf{Zitat \thesection.\arabic{ZitatNr}}  
  ...
```

Eingabe

```
\begin{Zitat}{Die Letzten ihrer Art}  
\label{ErstesZitat} ... \end{Zitat}
```

Ausgabe

Zitat 5.1 (Douglas Adams)

Das Tolle daran, die einzige zwischen richtig und falsch unterscheidende Art zu sein, ist, dass wir uns immer genau die Regeln ausdenken können, die uns gerade in den Kram passen.

(Quelle: Die Letzten ihrer Art)

Der Zugriff `\ref{ErstesZitat}` liefert 1.

Problembehebung

Abhilfe durch

```
\renewcommand{\theZitatNr}{  
  \thesection.\arabic{ZitatNr}}
```

Eigene Umgebungen mit Argumenten

Eingabe

```
\begin{Zitat}{Die Letzten ihrer Art}  
\label{ZweitesZitat} ... \end{Zitat}
```

Ausgabe

Zitat 5.2 (Douglas Adams)

*Noch sind wir zwar keine gefährdete Art, aber es ist nicht so,
dass wir nicht oft genug versucht hätten, eine zu werden.*

(Quelle: Die Letzten ihrer Art)

Der Zugriff `\ref{ZweitesZitat}` liefert 5.2.