



Die Ressourcenuniversität. Seit 1765.

Christian Degenkolb



Rechnen in \LaTeX

Eine kurze Vorstellung von **spreadtab**, **fp**
und **SageTeX**

15. Juli 2012



- Autor: Christian Tellechea (auch `chemfig`, `xstring`, ...)
- Version: 0.4b (2012/05/13)
- “[...] provides spreadsheet features for LaTeX table environments. [...] allows to construct tables in a manner similar to a spreadsheet [...] cells of a table have row and column indices [...] can be used in formulas to generate values in other cells [...]”
- kompatibel mit allen Tabellentypen, `booktabs`, `\multicolumn`, `\hline` und `\cline`
- <http://www.ctan.org/pkg/spreadtab>

Rechnen in Tabellen mit `spreadtab`

```
1 \usepackage{spreadtab}
2 ...
3 \begin{spreadtab}%
4 {{Tabellentyp}{Spaltendefinitionen}}
5 1                & 2                & Cell1:1 + Cell2:1\\
6 3                & 4                & Cell1:2 - Cell2:2\\
7 Cell1:1 / Cell2:1 & Cell1:2 * Cell2:2 & Cell3:1 ^ Cell3:2 \\
8 \end{spreadtab}
```

1	2	3
3	4	-1
0.333	8	0.333

Absolutes Referenzieren

- in der Form $\langle colref \rangle \langle rowref \rangle$ wobei $colref$ =[Buchstabe von a bis z] und $rowref$ =[Zahl von 1 startend]

```
1 1      & 2      & a1 + b1 \\
2 3      & 4      & a2 - b2 \\
3 a1/a2 & b1*b2 & c1^c2 \\
```

1	2	3
3	4	-1
0.333	8	0.333

Relatives Referenzieren

- in der Form $[x, y]$ wobei x =[horizontale Offset von der Zelle mit Formel] und y =[vertikale Offset von der Zelle mit Formel]

```
1 1      & 2      & [-2,0] + [-1,0] \\
2 3      & 4      & [-2,0] - [-1,0] \\
3 [0,-2] / [0,-1] & [0,-2] * [0,-1] & [0,-2] ^ [0,-1] \\
```

1	2	3
3	4	-1
0.333	8	0.333

Textzellen

- durch ein vorgestelltes @ definiert
- Zelle wird in Berechnungen ignoriert und ist nicht adressierbar
- mit `\renewcommand\STtextcell{<char>}` auf `<char>` änderbar
- leere Zellen oder Zellen mit nur Leerzeichen sind automatisch Textzellen

gemischte Zellen

- `<Textfeld> := {Numerisches Feld}<Ende des Textfeldes>`
- Trenner von `:=` durch `\renewcommand\STnumericfieldmarker{<text>}` auf `<text>` änderbar

```
1 1      & 2          \\
2 3      & b2= :={4} nicht 5 \\
3 a1+a2 & b1+b2      \\
```

```
1          2
3  b2= 4 nicht 5
4          6
```

- um Formeln nicht mehrmals eingeben zu müssen
- `\STcopy{>x,vy}{<Formel>}`
- `>x` kopiert die Formeln um x stellen nach rechts, nur `>` kopiert bis zum Ende der Zeile
- `vy` kopiert die Formeln um y nach unten, nur `v` kopiert bis zum Ende der Spalte
- Zellen mit belegtem numerischen Teil werden nicht überschrieben (A)
- Zellen mit Textinhalt, aber freiem numerischen Inhalt, werden “aufgefüllt” (B)

1	1	& \STcopy{v2}{a1 + a2} & \\	1	3		
2	\STcopy{v}{a1+1} &	& \\	2	5		
3		& B:={}	& \\	3	B102	
4	A:={99}	& \STcopy{>1}{a1 + a3} & \\	A99	4	105	

außerhalb der Tabelle

- `\STsavecell{<name>}{abs. Referenz}` als optionaler Parameter von `\begin{spreadtab}`

```
1 \begin{spreadtab}
2 [\STsavecell{\exp}{b3}]
3 ...
4 1 & 2 & \\
5 3 & b2= :={4} nicht 5 & \\
6 a1+a2 & b1+b2 & \\
7 ...
8 Inhalt von b3: \exp
```

1	2
3	b2= 4 nicht 5
4	6
Inhalt von b3: 6	

innerhalb der Tabelle

- in der Form `<<Reference>>`

```
1 @Verkauf & 5 \\
2 @Einkauf & 3 \\
3 @Profit (<<b1>>-<<b2>>) & b1-b2 \\
```

Verkauf	5
Einkauf	3
Profit (5-3)	2

mit `tag`, `cell` und `STtag`

- durch `tag(<name>)` wird `<name>` als Alias für Zelle festgelegt
- Zugriff innerhalb der Tabelle mit `cell(<name>)` ...
- ... und außerhalb der Tabelle mit `\STtag{<name>}`

```
1 1tag(eins) \  
2 4tag(vier) \  
3 cell(eins)+cell(vier) \  
4 a1*cell(vier)tag(erg) \  
5 ...  
6 a1 = \STtag{eins}  
7 Ergebnis = \STtag{erg}
```

$$\begin{array}{r} 1 \\ 4 \\ \hline 5 \\ 4 \\ a1 = 1 \\ \text{Ergebnis} = 4 \end{array}$$

runden von Werten

- `\STautoround{<n>}` bzw. `\STautoround*{<n>}` rundet auf n Nachkommastellen bzw. füllt bis auf n Nachkommastellen mit 0 auf

```
1 \STautoround{3}
```

```
2 ...
```

```
3 1    \\  
4 3    \\  
5 a1 / a2 \\  
5
```

```
1  
2  
3  
-----  
0.333
```

```
1 \STautoround*{3}
```

```
2 ...
```

```
3 1    \\  
4 3    \\  
5 a1 / a2 \\  
5
```

```
1.000  
3.000  
-----  
0.333
```

Dezimaltrennzeichen

- durch `\STsetdecimalsep{<char>}` auf *char* festlegen
- Standard ist Punkt

```
1 \STsetdecimalsep{,}
```

```
2 ...
```

```
3 1    & 2    & a1 + a2 \\  
4 3    & 4    & b1 - b2 \\  
5 a1/a2 & b1*b2 & c1^c2 \\  
5
```

```
1 2 | 3  
3 4 | -1  
-----  
0,333 8 | 0,333
```

Summen berechnen

- `\sum{< Intervall 1>;<Intervall 2>; ... ;<Intervall n>}`
- wobei ein Intervall eine einzelne Zelle, die absolut (a1) oder relativ ([2,1]) adressiert sein kann,...
- ... oder ein Rechteck aus Zellen, das durch die obere linke und unter rechte Zelle (a2:d5 oder [-1,-1]:[2,3]) begrenzt ist

```

1 1          & 9          \\
2 3          & 8          \\
3 \sum(a1; [0,-1]) & \sum(a1:b2) \\

```

$$\begin{array}{r}
 1 \quad 9 \\
 3 \quad 8 \\
 \hline
 4 \quad 21
 \end{array}$$

Fakultät berechnen

- `fact{<n>}` berechnet die Fakultät von n ($n \leq 18$)

```

1 1          & 3          \\
2 \fact(5) & \fact(b1) \\

```

$$\begin{array}{r}
 1 \quad 3 \\
 \hline
 120 \quad 6
 \end{array}$$

- `sumprod(<Intervall>, ...)` multipliziert Elemente und addiert die Produkte (Kap. 4.1.3)
- `rand()` / `randint(n,m)` zufällige Zahl zwischen $[0,1]$ / $[n,m]$ (Kap. 4.1.4)
- `gcd(n,m,...)` / `lcm(n,m,...)` größte gemeinsame Teiler / kleinste gemeinsame Teiler von n, m, \dots (Kap. 4.1.5)
- `scitodec(<Mantisse>EE<Exponent>)` wandelt von wissenschaftliche Notation (Mantisse * 10^{Exponent}) in Dezimal um (Kap. 4.1.6)
- weitere Makrofunktionen für Zellenwerte (Kap. 4.1.7) und Datumsumrechnungen (Kap. 4.3)

1	1 & <code>rand()</code> & <code>sumprod(a1:a2;b1:b2) \\</code>	1	0,12	12,12
2	3 & 4 & <code>gcd(a2,b2) \\</code>	3	4	1
3	@-3.1EE3 & <code>scitodec(a3) \\</code>	-3.1EE-3	-0,003	

- Autor: Michael Mehlich
- Version: jede Datei hat eigene Version, letztes Update 21. Okt. 1996
- “Fixed point arithmetic for TeX [...]”
- Einbinden mit `\usepackage[options]{fp}`
- <http://www.ctan.org/pkg/fp>

- `\FPiflt#1#2 <3> \else <4> \fi` Prüft $\#1 < \#2$ dann `<3>` ansonsten `<4>`
- `\FPifeg#1#2 <3> \else <4> \fi` Prüft $\#1 = \#2$ dann `<3>` ansonsten `<4>`
- `\FPifgt#1#2 <3> \else <4> \fi` Prüft $\#1 > \#2$ dann `<3>` ansonsten `<4>`
- `\FPifpos/neg/zero#1 <2> \else <3> \fi` Prüft $\#1 > 0 / < 0 / = 0$ dann `<2>`
- `\FPifint#1 <2> \else <3> \fi` Prüft ob $\#1$ Integer ist, dann `<2>` ansonsten `<3>`

```
1 \FPset\wertEins{1}
2 \FPset\wertDrei{3}
3 \FPiflt\wertEins\wertDrei 1 kleiner 3
   \else 1 nicht kleiner 3 \fi
4 \FPifpos\wertEins \FPset\wertTest{4}
   \else \FPset\wertTest{-4} \fi
5 wertTest=\FPprint\wertTest
```

1 kleiner 3
wertTest=4

Weitere nützliche Makros und Funktionen

- `\FPmin/max#1#2#3` #1 := min/max(#2,#3)
- `\FPln#1#2` #1 := ln(#2) `\FPpow#1#2#3` #1 := #2^{#3}
- `\FProot#1#2#3` #1 := #3 te Wurzel von #2
- `\FPsin/cos/tan#1#2` #1 := sin/cos/tan(#2)
- `\FPrandom#1` #1 := zufällige Zahl zwischen 0 und 1
- `\FPe` = Eulerzahl (2.718...) `\FPpi` = Pi (3.1415...)
- `\FPsolve#1#2#3` #1 := x mit $\#2*x+\#3=0$
- `\FPqsolve#1#2#3#4#5` #1,#2 := x mit $\#3*x^2+\#4*x+\#5=0$
- weitere siehe fp-Dokumentation

```
1 \FPmin\ergMin{7}{11}
2 \FPset\wertDrei{3}
3 \FPpow\ergPow\wertDrei{2}
4 \FPrandom\ergRand
5 \FPsolve\ergLSolv{4}{12}
```

```
ergMin= 7
ergPow=
8.999999999999999733
ergRand=
0.686674999392905737
ergLSolv= -3
```

- `\FPeval#1#2` #1 ist das Ergebnis der Berechnung #2
- mögliche Operationen in #2 sind:
+, -, *, /, ^, clip, trunc, round, sin, cos, tan, root, ln, min, max, ...
- per `\FPset` zugewiesene Makros werden direkt mit ihrem Namen angesprochen (ohne `\`)
- mit `\FPupn#1#2` ist auch die Umgekehrte Polnische Notation möglich

```
1 \FPeval\ergEins{(2*3 + 2) / 2 }
2 \FPset\wertNeun{9}
3 \FPeval\ergZwei{round(wertNeun / 8:2)}
4 \FPeval\ergDrei{trunc(tan(abs(-5)):3)}
5 \FPupn\ergUPN{3 4 + pi * 2 round}
```

```
ergEins=
4.000000000000000000
ergZwei= 1.13
ergDrei= -3.380
ergUPN= 21.99
```

- Autor: Dan Drake
- Version: (CTAN) 2.2.1 (Sage 5.1) v2.3.3 2012/01/16
- “[...] allows you to embed code, results of computations, and plots from the Sage mathematics software suite (<http://sagemath.org>) into \LaTeX documents. [...]”
- <http://www.ctan.org/pkg/sagetex>
- `sagetex.sty` aus $\$SAGEROOT/local/share/texmf/tex/generic/sagetex$ in \LaTeX -Suchpfad kopieren
- Einbinden mit `\usepackage{sagetex}`
- \LaTeX -Lauf, Sage-Lauf (generiert `<job>.sagetex.*`), \LaTeX -Lauf

Inline Sage-Code und sageblock

- `\sage{<sagecode>}` führt `<sagecode>` und gibt Ergebnisse aus
- `\begin{sageblock}<sagecode>\end{sageblock}` führt `<sagecode>` aus und setzt es in \LaTeX
- `\begin{sagesilent} ...` führt `<sagecode>` aus, aber setzt den Inhalt nicht in \LaTeX
- `\begin{sageverbatim} ...` setzt `<sagecode>` in \LaTeX , aber führt ihn nicht aus

```
1 Inline:  $2+2=\sage{2+2}$ 
2 mit sageblock
3 \begin{sageblock}
4   var('a,b')
5   eqn = [a+b==1, b-a==0]
6   s = solve(eqn, a,b)
7 \end{sageblock}
8 Lösung der  $\text{\textit{Gleichungen}}$ =
   \sage{eqn}:
9  $\sage{s}$ 
```

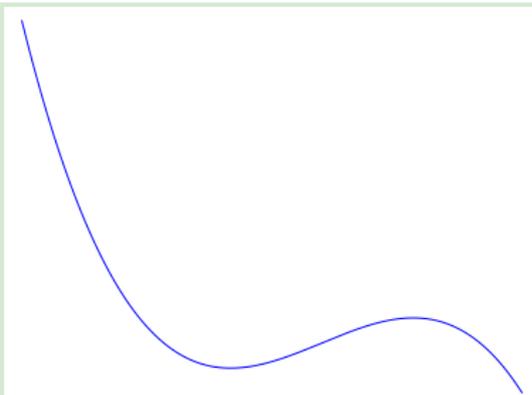
Inline: $2 + 2 = 4$
mit sageblock

```
var('a,b')
eqn = [a+b==1, b-a==0]
s = solve(eqn, a,b)
```

Lösung der Gleichungen =
 $[a + b = 1, -a + b = 0]$:
 $[[a = (\frac{1}{2}), b = (\frac{1}{2})]]$

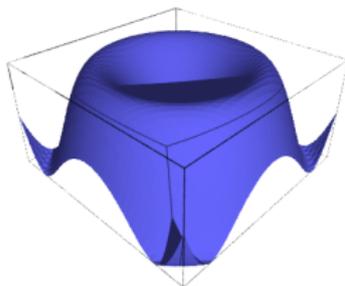
- `\sageplot`[<graphicx options>][<format>]{<graphic obj>,<args>}
plottet <graphic obj>
- <graphicx options> werden direkt an `\includegraphics` weitergereicht
- <format> Ausgabeformat, Standard ist EPS/PDF, möglich außerdem *png*

```
1 \begin{sagesilent}
2   var('x')
3   f(x) = -x^3+3*x^2+7*x-4
4 \end{sagesilent}
5 \sageplot[scale=.4]{plot(f, x,
   -5, 5), axes=False}
```

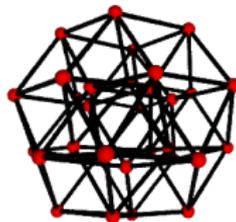


sageplot in 3D

```
1 \begin{sagesilent}
2   x, y = var('x y')
3 \end{sagesilent}
4 \sageplot[scale=.23]{
   plot3d(sin(pi*(x^2+y^2))/2,
   (x,-1,1),(y,-1,1))}
```

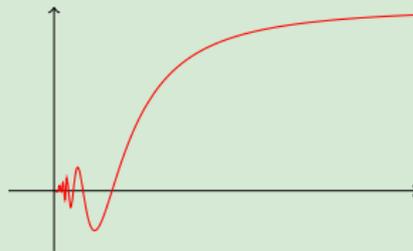
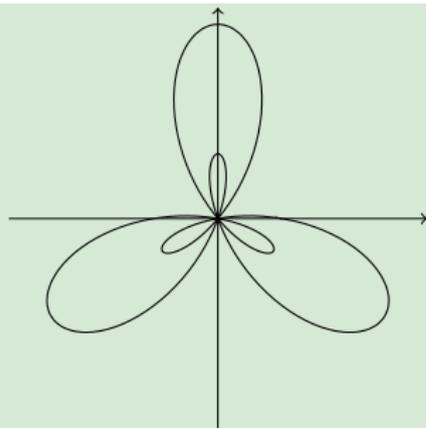
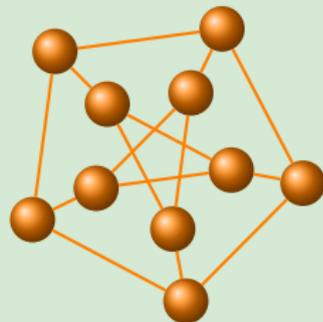


```
1 \begin{sagesilent}
2   G = graphs.CubeGraph(5)
3 \end{sagesilent}
4 \sageplot[scale=.23][png]{
   G.plot3d()}
```



sageplot mit TikZ

```
1 \usepackage{tkz-berge}
2 \begin{sagesilent}
3   g = graphs.PetersenGraph()
4   g.set_latex_options(tkz_style='Art')
5 \end{sagesilent}
6 \tikzstyle{every picture}=[rotate=45,
7   scale=1/2]
7 \sage{g}
```



- `\sagetexindent` Einrückungslänge der sageblock-Ausgabe
- `\sagetexpause/\sagetexunpause` Pausieren/Entpausieren der Sage-Berechnungen
- `\percent` Prozentzeichen (Sage kommt nicht mit `\%` klar und L^AT_EX nicht mit `%`)
- `makestatic.py infile [outfile]` ersetzt alle Sage-Aufrufe durch die Ergebnisse aus `<job>.sagetex.sout`
- `extractsagecode.py infile [outfile]` liest alle Sage-Aufrufe aus `infile` und speichert sie in `outfile`
- `remote-sagetex.py sagefile` führt `sagefile` auf entferntem System aus

`\endinput`