

# The drawmatrix package

Elmar Peise  
peise@aices.rwth-aachen.de

2016/12/20 v1.4.0

## Abstract

drawmatrix provides macros to visually represent matrices. Various options allow to change the visualizations, e.g., drawing rectangular, triangular, or banded matrices.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Drawing Matrices</b>	<b>2</b>
2.1	Size	3
2.2	Shape	3
2.2.1	Triangular and Trapezoidal Matrices	3
2.2.2	Banded Matrices	3
2.2.3	Diagonal Matrices	4
2.2.4	Super- and subscripts	4
2.3	Colors and Style	4
2.4	The Bounding Box	5
2.5	Coordinate system transformations	5
2.6	Position of the Label and Baseline	6
<b>3</b>	<b>Changing Defaults</b>	<b>6</b>
<b>4</b>	<b>Externalization</b>	<b>6</b>
<b>5</b>	<b>Implementation</b>	<b>7</b>
5.1	Package: TikZ	7
5.2	If for externalization	7
5.3	Key Declarations and Defaults	7
5.4	User Macros	10

# 1 Introduction

In many situations, visual representations of matrices facilitate the understanding of linear algebra properties, relations, and operations enormously. This package provides simple tools to bring such representations to L<sup>A</sup>T<sub>E</sub>X. For instance,

$$\boxed{A} \boxed{X} + \boxed{X} \boxed{B} = \boxed{C}$$

is typeset as follows:

```
\[
\drawmatrix[upper]A \;
\drawmatrix[width=.5]X +
\drawmatrix[width=.5]X \;
\drawmatrix[upper, size=.5, bbox height=1]B =
\drawmatrix[width=.5]C
\]
```

# 2 Drawing Matrices

`\drawmatrix` `\drawmatrix[<options>]{<label>}` draws a matrix labeled *<label>*: `\drawmatrix A` produces  $\boxed{A}$ . The *<options>*, which modify various aspects of drawn matrix through PGF's key-value system, are introduced in the following sections.

By default, the matrix is centered around its label, which is aligned with the surrounding text. The label is typeset in the surrounding mode and style.

`$\drawmatrix A$:`  $\boxed{A}$   
`{\bf \drawmatrix A}:`  $\boxed{\mathbf{A}}$   
`{\large \drawmatrix A}:`  $\boxed{A}$

In equations, parentheses (spanned with `\left` and `\right`), subscripts, and superscripts naturally extend to the drawn shape:  $\left(\boxed{A}_i + \boxed{B}^{-1}\right)\boxed{C}$ .

Used in matrix products such as  $\boxed{A} \boxed{B}$ , a little space (`\;`) helps to yield a more natural result:  $\boxed{A} \boxed{B}$ .

`label text` Note that the *<label>* is stored in `label text`, which can also be set directly to overwrite *<label>*.

`\drawmatrix[label text=B]A$:`



## 2.1 Size

By default, matrices are of size  $1 \times 1$  in terms of *TikZ* units. The width and height of a matrix are set through, respectively, `width= $\langle dimension \rangle$`  and `height= $\langle dimension \rangle$` . A width or height of 0 are useful to represent vectors:

`\drawmatrix[width=0]A:`



`size= $\langle dimension \rangle$`  sets both the width and height to  $\langle dimension \rangle$ , resulting in a square matrix.

## 2.2 Shape

By default matrices are rectangular.

### 2.2.1 Triangular and Trapezoidal Matrices

Lower and upper triangular matrices are obtained by, respectively, setting the keys `lower` and `upper`. Hereby, non-square matrices become trapezoidal.

`\drawmatrix[lower]L:`



`\drawmatrix[upper, width=1.5]U:`



### 2.2.2 Banded Matrices

Matrices are drawn as banded with the key `banded`. The bandwidth, i.e., the horizontal/vertical extent from the diagonal, is set by `bandwidth= $\langle dimension \rangle$`  (default: 0.3);


`\drawmatrix[banded]B:`




`\drawmatrix[bandwidth=.5]B:`





Banding for the lower and upper part of the matrices can be specified separately through `lower banded` and `upper banded`. Separate bandwidths are set through `lower bandwidth= $\langle dimension \rangle$`  and `upper bandwidth= $\langle dimension \rangle$` :

`\drawmatrix[lower banded]B:` 


`\drawmatrix[lower bandwidth=.5, upper bandwidth=.2]B:` 

Banding on rectangular matrices applies to the smaller of the two dimensions:

`\drawmatrix[banded, width=.8]B:` 


`\drawmatrix[upper banded, height=.7]B:` 

**banded** can be combined with **lower** or **upper** to draw the intersection of both shapes.

`\drawmatrix[banded, lower]L:` 


### 2.2.3 Diagonal Matrices


**diag** `diag` is a shorthand for **banded** with **bandwidth=0**:


`\drawmatrix[diag]D:` 

### 2.2.4 Super- and subscripts


**label base** `label base` defines the label to be centered in the `drawmatrix`, the actual label will be aligned to this label at the. The default alignment is at the **base west** of the label, which can be changed through the `label anchor` key. This feature is useful to, e.g., draw centered labels with exponents:

`$\drawmatrix[size=.5]{A^T}$:` 

`$\drawmatrix[size=.5, label base=A]{A^T}$:` 

`$\drawmatrix[size=.5, label base=A, label base anchor=base east]{^0_1A}$:` 

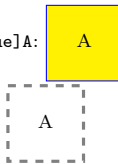
**exponent** `exponent` is a shortcut to add an exponent to matrix without offsetting the label. Internally, it sets the `label base` to the current `label text` and adds the `exponent` of the `label text`.

`$\drawmatrix[size=.5, exponent=T]A$:` 

## 2.3 Colors and Style

By default, matrices are drawn in gray and filled white. The `TikZ` keys `draw=<color>` and `fill=<color>` change these colors. In fact, all keys not recognized by this package are passed to the `TikZ \filldraw` command drawing the matrix.


```
\drawmatrix[fill=yellow, draw=blue]A:
\drawmatrix[very thick, dashed]A:
```



## 2.4 The Bounding Box

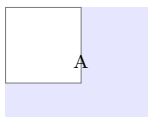
All matrices are contained in a rectangular bounding box. To draw this bounding box (e.g., to visualize the 0 entries in the matrix), use `bbbox style={\langle style \rangle}`; this style is applied to the TikZ `\node` that is the bounding box.

```
\drawmatrix[lower, bbbox style={fill=blue!10}]L:
```



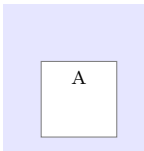
By default, the bounding box is just large enough to contain the matrix. Its size is changed through the keys `bbbox height=\langle dimension \rangle` and `bbbox width=\langle dimension \rangle` (or `bbbox size=\langle dimension \rangle` to set them both). The label of the matrix (and thus the alignment with respect to the surrounding text) are fixed at the center of the bounding box, while the matrix is positioned at its top-left corner.

```
\drawmatrixset{bbbox style={fill=blue!10}}
\drawmatrix[bbbox width=2, bbbox height=1.5]A:
```



The matrix can be positioned within its bounding box through `offset height=\langle dimension \rangle` and `offset width=\langle dimension \rangle` (or just `offset=\langle dimension \rangle` to shift along the diagonal).


```
\drawmatrixset{bbbox style={fill=blue!10}}
\drawmatrix[bbbox size=2, offset width=.5, offset height=.75]A:
```



## 2.5 Coordinate system transformations


`scale=\langle factor \rangle` scales all dimensions passed to a matrix:

```
\drawmatrix[scale=.6]A \drawmatrix[scale=.6, width=.5]B:
```



`x=\langle value \rangle` and `y=\langle value \rangle` define the coordinate system for all unit-less dimensions.

```
\drawmatrix[x=.6cm, y=.4cm]A \drawmatrix[x=.6cm, y=.4cm, width=1cm]B:
```



## 2.6 Position of the Label and Baseline

By default, the label's `mid` is positioned at the bounding box's `center` and its `base` is used as the whole drawing's baseline. This is controlled by the keys `label anchor`, `label pos`, and `baseline`. Here, `<position>` has to be an anchor of one of the following nodes: `bbox` (the bounding box), `matrix` (the matrix itself), or `label` (the label).

`label anchor`  
`label pos`  
`baseline`

```
\drawmatrixset{bbox height=1, height=.5, bbox style={fill=blue!10}}
```

```
\drawmatrix[label pos=bbox.south, label anchor=south]A: A
```

```
\drawmatrix[label pos=matrix.north west]A: A
```

```
\drawmatrix[baseline=label.north]A: A
```

```
\drawmatrix[baseline=bbox.south]A: A
```

## 3 Changing Defaults

`\drawmatrixset` Specifying `<options>` with `\drawmatrixset{<options>}` applies them to all following uses of `\drawmatrix` within the current scope.

```
\drawmatrixset{height=.5, lower}
```

```
$\drawmatrix A \; \drawmatrix B$:
```



Furthermore, TikZ keys for the entire picture, the bounding box, the matrix itself and the label can be set through the styles `every bbox`, `every drawmatrix`, and `every label`.

`every bbox`  
`every drawmatrix`  
`every label`

```
\drawmatrixset{every drawmatrix/.append style={rounded corners=5pt}}
```

```
$\drawmatrix A \; \drawmatrix[lower]B$:
```



## 4 Externalization

`\drawmatrix` behaves as any other TikZ picture, therefore when externalization is enabled, all matrix visualizations are also externalized. However, since there are usually many `\drawmatrix` pictures, each of which is very small and fast to produce, their externalization would mean a tremendous overhead. To avoid this overhead without explicitly dis- and re-enabling externalization throughout the document, `externalize=false` disables externalization for all `\drawmatrix` pictures:

`externalize`

```
\drawmatrixset{externalize=false}
```

## 5 Implementation

This section describes the implementation details of the `drawmatrix` package.

### 5.1 Package: `TikZ`

The `tikz` package is used for drawing.

```
1 \RequirePackage{tikz}
```

### 5.2 If for externalization

`\TeX` if representing whether to explicitly disable `TikZ` externalization.

```
\ifdrawmatrix@externalize
```

```
2 \newif\ifdrawmatrix@externalize
```

### 5.3 Key Declarations and Defaults

We rely on PGF keys as much as we can.

```
3 \pgfkeys{
```

```
    Everything happens in the path /drawmatrix.
```

```
4     drawmatrix/.is family,
```

```
5     drawmatrix/.cd,
```

`picture` `picture` is the style for the `\tikzpicture` in which the matrix is drawn. `baseline` sets the baseline of the picture to a named coordinate of the matrix (default: base of the label).

```
6     picture/.style={},
```

```
7     path/.style={},
```

```
8     baseline/.style={picture/.append style={baseline=(drawmatrix #1)}},
```

```
9     scale/.style={path/.append style={scale=#1}},
```

```
10    x/.style={path/.append style={x=#1}},
```

```
11    y/.style={path/.append style={y=#1}},
```

```
12    baseline=label.base,
```

`bbox` `bbox` is the style of the bounding box, to which `bbox style` appends keys.

`bbox style` `bbox style` `bbox/.style={}`,

```
14    bbox style/.style={bbox/.append style={#1}},
```

`bbox height` `bbox height` and `bbox width` don't have default values. `bbox size` sets them both to the same value.

`bbox size` `bbox height/.initial`,

```
16    bbox width/.initial,
```

```
17    bbox size/.style={bbox height=#1, bbox width=#1},
```

offset height offset height and offset width are 0 by default. offset sets them both to the same value.

offset width

```

offset 18 offset height/.initial=0,
      19 offset width/.initial=0,
      20 offset/.style={offset height=#1, offset width=#1},

```

height width and height are 1 (TikZ unit) by default. size sets them both to the same value.

width

```

size 21 height/.initial=1,
     22 width/.initial=1,
     23 size/.style={height=#1, width=#1},

```

lower bandwidth The lower bandwidth and upper bandwidth don't have default values.

upper bandwidth bandwidth sets them both to the same value.

bandwidth

```

bandwidth 24 lower bandwidth/.initial,
          25 upper bandwidth/.initial,
          26 bandwidth/.style={lower bandwidth=#1, upper bandwidth=#1},

```

lower banded lower banded and upper banded are shortcuts to set the corresponding band-

upper banded widths to the default value of 0.3 (TikZ units). banded sets them both.

banded

```

banded 27 lower banded/.style={lower bandwidth=.3},
       28 upper banded/.style={upper bandwidth=.3},
       29 banded/.style={lower banded, upper banded},

```

lower lower and upper are implemented by setting the opposite bandwidth to 0. diag

upper sets them both.

diag

```

diag 30 lower/.style={upper bandwidth=0},
     31 upper/.style={lower bandwidth=0},
     32 diag/.style={lower, upper},

```

label text label is the style for the label with the text label text. label pos sets the

label label at a named coordinate of the matrix (default: center of the bounding box).

label pos label anchor sets the label's anchor (default: in the middle).

label anchor

```

label anchor 33 label text/.initial,
            34 label/.style={},
            35 label pos/.style={label/.append style={at=(drawmatrix #1)}}},
            36 label pos=bbbox.center,
            37 label anchor/.style={label/.append style={anchor=#1}},
            38 label anchor=mid,

```

label base label base and label base anchor allow to offset labels with exponents.

label base anchor

```

label base anchor 39 label base/.initial,
                 40 label outer/.style={},
                 41 label base anchor/.style={label outer/.append style={
                 42 anchor=#1, at=(drawmatrix label.#1)
                 43 }},
                 44 label base anchor=base west,

```



**exponent** **exponent** is a shortcut to add an exponent to the label text without using the **label base**.

```

45   exponent/.style={
46       label base/.expanded=\pgfkeysvalueof{/drawmatrix/label text},
47       label text/.append=^{\#1}
48   },

```

Unknown keys are collected in `/drawmatrix/drawmatrix`.

```

49   drawmatrix/.style={},
50   .unknown/.code={%
51       \let\dm@currname\pgfkeyscurrentname%
52       \let\dm@currval\pgfkeyscurrentvalue%
53       \ifx#1\pgfkeysnovalue\pgfkeysalso{
54           drawmatrix/.append style/.expand once={\dm@currname}
55       }\else\pgfkeysalso{
56           drawmatrix/.append style/.expand twice={%
57               \expandafter\dm@currname\expandafter=\dm@currval%
58           }
59       }\fi%
60   },

```

**every picture** The default style for matrices: **every picture** applies to all `\tikzpictures`  
**every bbox** the matrices are drawn in, **every bbox** applies to all bounding boxes,  
**every drawmatrix** **every drawmatrix** applies to the matrices themselves, and **every label** applies  
**every label** to the labels.

```

61   every picture/.style={},
62   every bbox/.style={
63       name=drawmatrix bbox,
64       inner sep=0
65   },
66   every drawmatrix/.style={
67       fill=white,
68       draw=gray
69   },
70   every label/.style={
71       name=drawmatrix label,
72       outer sep=0,
73       inner sep=0
74   },
75   every node/.style={
76       name=drawmatrix matrix,
77       outer sep=0,
78       inner sep=0,
79       anchor=north west,
80       at=(drawmatrix north west)
81   },

```

**externalize** **externalize** sets a  $\TeX$  if (default: `true` = behave as all pictures).

```

82   externalize/.is if=drawmatrix@externalize,

```

```

83   externalize=true
84 }

```

## 5.4 User Macros

`\drawmatrixset` as a simple shortcut like `\tikzset`.

```
\drawmatrixset
```

```
85 \newcommand\drawmatrixset[1]{\pgfqkeys{/drawmatrix}{#1}}
```

Here we go, the main thing: `\drawmatrix`. First, apply the options and extract the sizes from the PGF keys.

```
\drawmatrix
```

```

86 \newcommand\drawmatrix[2][]{\{%
87   \drawmatrixset{
88     label text={#2},
89     #1,
90     label text/.get=\dm@labeltext,
91     height/.get=\dm@height,
92     width/.get=\dm@width,
93     lower bandwidth/.get=\dm@lowerbandwidth,
94     upper bandwidth/.get=\dm@upperbandwidth,
95     offset height/.get=\dm@offsetheight,
96     offset width/.get=\dm@offsetwidth,
97     bbox height/.get=\dm@bboxheight,
98     bbox width/.get=\dm@bboxwidth,
99     label base/.get=\dm@labelbase
100   }%

```

Prepare the label text and, if needed label outer text (for alignment). This needs to be outside the tikzpicture to properly detect math mode.

```

101   \ifmmode\edef\dm@labeltext{\$dm@labeltext\$}\fi%
102   \expandafter\ifx\dm@labelbase\pgfkeysnovalue\else%
103     \let\dm@labeltextouter\dm@labeltext%
104     \edef\dm@labelbase{%
105       \ifmmode\$dm@labelbase\$else\dm@labelbase\fi%
106     }%
107     \def\dm@labeltext{\phantom{\dm@labelbase}}%
108   \fi%

```

Disable externalization if `externalize=false`. Start the picture.

```

109   \ifdrawmatrix@externalize\else%
110     \ifx\tikz@library@external@loaded\undefined\else%
111       \tikzset{external/export=false}%
112     \fi%
113   \fi%
114   \begin{tikzpicture}[/drawmatrix/every picture, /drawmatrix/picture]

```

Parse width, height, the minimum dimension and zero for comparison purposes.

```
115     \path[/drawmatrix/path] (\dm@width, \dm@height);

```

```

116 \pgfgetlastxy\dm@width\dm@height
117 \path[/drawmatrix/path] (\dm@offsetwidth, \dm@offsetheight);
118 \pgfgetlastxy\dm@offsetwidth\dm@offsetheight
119 \pgfmathsetlengthmacro\dm@minsize{min(\dm@width, \dm@height)}
120 \pgfmathsetlengthmacro\dm@zero{0.0}

```

Prepare the band widths: First, if the matrix is not banded, the bandwidth is set to the smaller matrix dimension. Then, the band width is limited by this smaller dimension.

```

121 \expandafter\ifx\dm@lowerbandwidth\pgfkeysnovalue
122 \def\dm@lowerbandwidth{\dm@minsize}
123 \else
124 \path[/drawmatrix/path] (\dm@lowerbandwidth, 0);
125 \pgfgetlastxy\dm@lowerbandwidth\dm@zero
126 \fi
127 \expandafter\ifx\dm@upperbandwidth\pgfkeysnovalue
128 \def\dm@upperbandwidth{\dm@minsize}
129 \else
130 \path[/drawmatrix/path] (0, \dm@upperbandwidth);
131 \pgfgetlastxy\dm@zero\dm@upperbandwidth
132 \fi
133 \pgfmathsetlengthmacro\dm@lowerbandwidth{
134 min(\dm@minsize, \dm@lowerbandwidth)
135 }
136 \pgfmathsetlengthmacro\dm@upperbandwidth{
137 min(\dm@minsize, \dm@upperbandwidth)
138 }

```

Set the default bounding box size.

```

139 \expandafter\ifx\dm@bboxheight\pgfkeysnovalue
140 \pgfmathsetlengthmacro\dm@bboxheight{
141 \dm@height + \dm@offsetheight
142 }
143 \else
144 \path[/drawmatrix/path] (0, \dm@bboxheight);
145 \pgfgetlastxy\dm@zero\dm@bboxheight
146 \fi
147 \expandafter\ifx\dm@bboxwidth\pgfkeysnovalue
148 \pgfmathsetlengthmacro\dm@bboxwidth{
149 \dm@width + \dm@offsetwidth
150 }
151 \else
152 \path[/drawmatrix/path] (\dm@bboxwidth, 0);
153 \pgfgetlastxy\dm@bboxwidth\dm@zero
154 \fi

```

Reset the bounding box and begin with (drawing) the path for the bounding box.

```

155 \pgfresetboundingbox
156 \node[/drawmatrix/every bbox, /drawmatrix/bbox,
157 minimum height=\dm@bboxheight,

```

```
158         minimum width=\dm@bboxwidth] {};
```

Whether needed or not, declare all matrix corners.

```
159     \path (drawmatrix bbox.north west)
160         ++(\dm@offsetwidth, -\dm@offsetheight)
161         ++(.5\pgflinewidth, -.5\pgflinewidth)
162         coordinate (drawmatrix north west)
163         ++(\dm@width, 0)
164         +(-\dm@minsize + \dm@upperbandwidth, 0)
165         coordinate (drawmatrix north)
166         +(0, -\dm@minsize + \dm@upperbandwidth)
167         coordinate (drawmatrix east)
168         ++(0, -\dm@height)
169         coordinate (drawmatrix south east)
170         ++(-\dm@width, 0)
171         +(\dm@minsize - \dm@lowerbandwidth, 0)
172         coordinate (drawmatrix south)
173         +(0, \dm@minsize - \dm@lowerbandwidth)
174         coordinate (drawmatrix west);
```

Add an invisible node the size of the matrix.

```
175     \node[/drawmatrix/every node,
176         minimum height=\dm@height,
177         minimum width=\dm@width] {};
```

Now, draw only what is needed of the matrix. Otherwise path modifications (e.g., such as rounded corners) might not work.

```
178     \filldraw[/drawmatrix/every drawmatrix, /drawmatrix/drawmatrix]
179         (drawmatrix north west)
180         \ifx\dm@upperbandwidth\dm@zero
181             \ifx\dm@width\dm@minsize\else -- (drawmatrix north) \fi
182             \ifx\dm@height\dm@minsize\else -- (drawmatrix east) \fi
183         \else
184             -- (drawmatrix north)
185             \ifx\dm@upperbandwidth\dm@minsize\else
186                 -- (drawmatrix east)
187             \fi
188         \fi
189         -- (drawmatrix south east)
190         \ifx\dm@lowerbandwidth\dm@zero
191             \ifx\dm@width\dm@minsize\else -- (drawmatrix south) \fi
192             \ifx\dm@height\dm@minsize\else -- (drawmatrix west) \fi
193         \else
194             -- (drawmatrix south)
195             \ifx\dm@lowerbandwidth\dm@minsize\else
196                 -- (drawmatrix west)
197             \fi
198         \fi
199         -- cycle;
```

The label.

```

200     \node[/drawmatrix/every label, /drawmatrix/label]
201         {\dm@labeltext};
202     \expandafter\ifx\dm@labelbase\pgfkeysnovalue\else
203         \node[/drawmatrix/every label, /drawmatrix/label,
204             /drawmatrix/label outer] {\dm@labeltextouter};
205     \fi
206 \end{tikzpicture}%
207 }}

```

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

<b>B</b>		<b>O</b>
\banded . . . . .	<i>3, 27</i>	\offset . . . . . <i>5, 18</i>
\bandwidth . . . . .	<i>3, 24</i>	\offset_height . . . <i>5, 18</i>
\baseline . . . . .	<i>6, 6</i>	\offset_width . . . <i>5, 18</i>
\bbox . . . . .	<i>13</i>	<b>H</b>
\bbox_height . . . . .	<i>5, 15</i>	\height . . . . . <i>3, 21</i>
\bbox_size . . . . .	<i>5, 15</i>	<b>I</b>
\bbox_style . . . . .	<i>5, 13</i>	\ifdrawmatrix@externalize
\bbox_width . . . . .	<i>5, 15</i>	. . . . . <i>2</i>
<b>D</b>	<b>L</b>	<b>P</b>
\diag . . . . .	<i>4, 30</i>	\pgfkeysvalueof . . . <i>46</i>
\drawmatrix . . . . .	<i>2, 86</i>	\picture . . . . . <i>6</i>
\drawmatrixset . . . . .	<i>6, 85, 87</i>	<b>S</b>
<b>E</b>	<b>U</b>	\size . . . . . <i>3, 21</i>
\every_bbox . . . . .	<i>6, 61</i>	\upper . . . . . <i>3, 30</i>
\every_drawmatrix . . . . .	<i>6, 61</i>	\upper_banded . . . <i>3, 27</i>
\every_label . . . . .	<i>6, 61</i>	\upper_bandwidth . . <i>3, 24</i>
\every_picture . . . . .	<i>61</i>	<b>W</b>
	\label . . . . . <i>33</i>	\width . . . . . <i>3, 21</i>
	\label_anchor . . . . <i>6, 33</i>	
	\label_base . . . . . <i>4, 39</i>	
	\label_base_anchor . . <i>4, 39</i>	
	\label_pos . . . . . <i>6, 33</i>	
	\label_text . . . . . <i>2, 33</i>	
	\lower . . . . . <i>3, 30</i>	
	\lower_banded . . . . <i>3, 27</i>	
	\lower_bandwidth . . <i>3, 24</i>	

## Change History

v1.0.0	linewidth/2 offset . . . . .	<i>1</i>
General: Initial Version . . . . .	<i>1</i>	v1.1.1
v1.0.1	General: Bugfix: Remove extra space after vectors . . . . .	<i>1</i>
General: Bugfix: Collapsible bbox (label placement for vectors) . .	<i>1</i>	v1.2.0
v1.0.2	General: Added coordinate transformations . . . . .	<i>1</i>
General: Bugfix: Bbox had a		

v1.3.0		v1.4.0	
General: Added label base (shifted		General: Added exponenet	
exponents) . . . . .	1	shortcut . . . . .	1