

Befehlsübersicht für PSTricks

Uwe Siart

Version 0.67 vom 16. Mai 2005

1 Grundeinstellungen

Bildumgebung

```
\begin{pspicture}[base](x1,y1)(x2,y2)
\end{pspicture}
```

Einheitslängen

```
xunit = dim    runit = dim
yunit = dim    unit = dim
```

Mit `unit` werden alle Werte gleichzeitig gesetzt.

Farbdefinition

```
\definecolor[class]{name}{model}{spec}
\colorlet{name}[model]{expr}
```

Dabei sind *spec* eine kommaseparierte Werteliste und *expr* ein Farbausdruck. Mit der Option `[dvipsnames]` werden die Crayola-Farben des `dvips` verfügbar. Weitere Informationen liefert die Dokumentation des Paketes `xcolor`¹.

Globale Parameter setzen

```
\psset{key1=value1, key2=value2, ...}
\newpsstyle{name}{key1=value1, ...}
```

Globale Parameter können lokal durch `[par]` oder `[style = name]` übersteuert werden.

Vollwinkelteilung

```
\degrees[div]
\degrees
\radians
```

Koordinatendarstellung umschalten

```
\NormalCoor    \SpecialCoor
```

Mögliche Koordinatendarstellungen (`\SpecialCoor`)

```
(x,y)    (!code) ([par]node)
(r;a)    (node)  (coor1|coor2)
```

2 Grafische Grundelemente

Linien, Polygone, Rahmen

```
\psline[par]{ends}(x0,y0)(x1,y1)...
\pspolygon[par](x0,y0)(x1,y1)...
\psframe[par](x0,y0)(x1,y1)
\psdiamond[par](x0,y0)(x1,y1)
\pstriangle[par](x0,y0)(x1,y1)
```

Bei `\psdiamond` und `\pstriangle` geben (x_0, y_0) das Zentrum und (x_1, y_1) die Breite und Höhe an. Eine Drehung um das Zentrum gibt der Parameter `gangle` an.

Kreisbögen

```
\psarc[par]{ends}(x,y){rad}{ang}{ang}
\psarcn[par]{ends}(x,y){rad}{ang}{ang}

\pscircle[par](x,y){rad}
\pswedge[par](x,y){rad}{ang}{ang}
\psellipse[par](x,y)(a,b)
```

Parameter für Linien

```
linewidth = dim
linecolor = color
linestyle = style
lineararc = dim
dash = dim1 dim2
dotsep = dim
border = dim
bordercolor = color
doubleline = true/false
doublesep = dim
doublecolor = color
arrows = ends
```

Mögliche Werte für `linestyle`

none	solid	dashed	dotted
------	-------	--------	--------

Mögliche Werte für `arrows`

<->	>-<	<<->>	>>-<<	-
*- *	[-]	(-)	o-o	***
oo-oo	***	c-c	cc-cc	C-C

¹Seit [2004/05/12 v0.21] lädt `pstricks.sty` das Paket `xcolor`. Die Verwendung der älteren Pakete `color` oder `pstcol` ist aber weiterhin möglich.

Linienenden können auch in der Form `{ends}` in Linienbefehlen angegeben werden.

Parameter für Linienenden

```
arrowsize = dim num
arrowlength = num
arrowinset = num
tbarsize = dim num
bracketlength = num
```

Punkt an jeder Koordinate

```
\psdot*[par] (x0,y0)
\psdots*[par] (x0,y0) (x1,y1) ...
```

Parameter für Punkte ('var' bedeutet optional)

```
dotstyle = style
dotsize = dim 'num'
dotstyle = num1 'num2'
dotangle = ang
```

Mögliche Werte für dotstyle

*	o	+	x
asterisk	oplus	otimes	
square	diamond	triangle	pentagon
square*	diamond*	triangle*	pentagon*

Kurven

```
\psbezier[par] (x0,y0) ... (x3,y3)
\pscurve[par] (x0,y0) (x1,y1) ...
\psecurve[par] (x0,y0) (x1,y1) ...
\psccurve[par] (x0,y0) (x1,y1) ...
```

3 Textboxen

Rahmenboxen

```
\psframebox[par]{stuff}
\psdblframebox[par]{stuff}
\psshadowbox[par]{stuff}
\pscirclebox[par]{stuff}
\psovalbox[par]{stuff}
\psdiabox[par]{stuff}
\pstribox[par]{stuff}
```

Parameter für Rahmen und geschlossene Pfade

```
fillstyle = style
fillcolor = color
framearc = num
framesep = dim
hatchwidth = dim
hatchcolor = color
hatchangle = ang
hatchsep = dim
cornersize = relative/absolute
dimen = inner/middle/outer
```

Mögliche Werte für fillstyle

```
none solid vlines vlines*
hlines hlines* crosshatch crosshatch*
```

Die Verwendung der *-Versionen bei Rahmen und Kurven wirkt wie `fillstyle = solid`. Die verwendete Füllfarbe ist in diesem Fall `linecolor`.

4 Platzierung von Objekten

Verschiebung und Drehung

```
\rput[ref]{rot}(x,y){stuff}
\multirput[ref]{rot}(x,y)(a,b){rep}{stuff}
```

Parameter für eine Nullpunktverschiebung

```
origin = {coord}
```

Mögliche Werte für ref

<i>horizontal</i>	<i>vertikal</i>
l	t
r	b
	B

Beschriftung von Punkten

```
\uput{sep}[ang]{rot}(x,y){stuff}
```

Parameter für Beschriftungen

```
labelsep = dim
```

5 Knoten und Knotenverbindungen

Knoten festlegen (pst-node)

```
\rnode[ref]{name}{stuff}
\Rnode(x,y){name}{stuff}
\pnode(x,y){name}
\cnode[par](x,y){rad}{name}
\Cnode[par](x,y){name}
\circlenode[par]{name}{stuff}
\cnodeput[par]{ang}(x,y){name}{stuff}
\ovalnode[par]{name}{stuff}
\dotnode[par](x,y){name}
```

Knotenverbindungen (Segmentanzahl in Klammern)

```
\nccurve[par]{arrows}{nodeA}{nodeB} (0)
\ncline[par]{arrows}{nodeA}{nodeB} (1)
\ncarc[par]{arrows}{nodeA}{nodeB} (1)
\nccircle[par]{arrows}{nodeA}{nodeB} (1)
\ncdiag[par]{arrows}{nodeA}{nodeB} (2)
\ncdiag[par]{arrows}{nodeA}{nodeB} (3)
\ncbar[par]{arrows}{nodeA}{nodeB} (3)
\ncangle[par]{arrows}{nodeA}{nodeB} (3)
\ncangles[par]{arrows}{nodeA}{nodeB} (4)
\ncloop[par]{arrows}{nodeA}{nodeB} (5)
```

Punktverbindungen (Segmentanzahl in Klammern)

```
\pccurve[par]{arrows}(x1,y1)(x2,y2) (0)
\pcline[par]{arrows}(x1,y1)(x2,y2) (1)
\pcarc[par]{arrows}(x1,y1)(x2,y2) (1)
\pcdiag[par]{arrows}(x1,y1)(x2,y2) (2)
\pcdiag[par]{arrows}(x1,y1)(x2,y2) (3)
\pcbar[par]{arrows}(x1,y1)(x2,y2) (3)
\pcangle[par]{arrows}(x1,y1)(x2,y2) (3)
\pcangles[par]{arrows}(x1,y1)(x2,y2) (4)
\pcloop[par]{arrows}(x1,y1)(x2,y2) (5)
```

Parameter für Knoten und Verbindungen

```
ncurv = num      arcangle = ang
offset = dim      loopsize = dim
arm = dim        [XY]nodesep = dim
angle = ang      radius = dim
```

Die Parameter `[XY]nodesep`, `offset`, `arm` und `angle` können in den Varianten `parA` und `parB` auch für beide Endknoten separat gesetzt werden. Zusätzlich gelten alle Linienparameter. Die Verschiebung `offset` zählt positiv zur linken Seite des Pfades. `radius` dient als globaler Parameter für `\Cnode`.

Knoten beschriften

```
\nput[par]{ang}{node}{stuff}
```

Knoten- und Punktverbindungen beschriften

```
\ncput[par]{stuff}
\naput[par]{stuff}
\nbput[par]{stuff}
```

Die Befehle `\lput`, `\aput`, `\bput`, `\Aput`, `\Bput`, `\Lput`, `\Mput` und `\Rput` sind `obsolete`, werden aber weiter unterstützt.

Parameter für Beschriftungen

```
nrot = rot
npos = num
```

Mit der Angabe `nrot=:ang` erfolgt die Drehung relativ zur Richtung der Verbindungslinie (häufig `nrot=:U`).

6 Gitter und Achsenkreuze

Gitterbefehl

```
\psgrid(x0,y0)(x1,y1)(x2,y2)
```

Gitterparameter

```
gridwidth = dim
gridcolor = color
griddots = num
gridlabels = dim
gridlabelcolor = color
subgriddiv = int
subgridwidth = dim
```

```
subgridcolor = color
subgriddots = num
```

Achsenkreuz

```
\psaxes(x0,y0)(x1,y1)(x2,y2)
```

Parameter für Achsenkreuze

```
ticks = x/y/all/none
labels = x/y/all/none
tickstyle = full/top/bottom
ticksize = dim
showorigin = false/true
axesstyle = axes/frame/none
labelsep = dim
```

```
Ox = num   Dx = num   dx = num
Oy = num   Dy = num   dy = num
```

`Ox` und `Oy` sind die Startwerte der Nummerierungen im Ursprung. `Dx` und `Dy` sind die Inkremente der Nummerierungen. `dx` und `dy` sind die Abstände der Achsenmarken.

Stil der Achsenbezeichnungen

```
\renewcommand{\pshlabel}[1]{commands#1}
\renewcommand{\psvlabel}[1]{commands#1}
```

7 Erweiterungen und Verschiedenes

Zusammenhängende und geschlossene Pfade

```
\pscustom[par]{paths}
```

Grafiken begrenzen (clipping)

```
\psclip{clipobjects}
...
\endpsclip
```

Schatten

```
shadow = true/false
shadowsize = dim
shadowangle = ang
shadowcolor = color
```

Rahmen um Knoten herum

```
\ncbox[par]{nodeA}{nodeB}
\ncarcbox[par]{nodeA}{nodeB}
```

Parameter für zusammenhängende Pfade

```
linetype = int
liftpen = 0/1/2
```

Daten einlesen (pst-plot)

```
\readdata{object}{filename}
\savedata{object}{filename}
```

Daten plotten

```
\fileplot[par]{filename}
\dataplot[par]{object}
\listplot[par]{object}
```

Funktionen plotten

```
\psplot{x1}{x2}{y(x)}
\parametricplot{t1}{t2}{x(t) y(t)}
```

Plotparameter

```
plotstyle = style
plotpoints = int
```

Mögliche Werte für plotstyle

```
dots      line      polygon
curve     ecurve    ccurve
```

Einige POSTSCRIPT®-Operatoren

```
add      sub      mul      div
abs      neg      mod      dup
sin      cos      atan      sqrt
exp      ln       log      exch
ceiling  floor     round   truncate
```

Die trigonometrischen Funktionen erwarten ihr Argument in Grad.

Wichtige Zahlenwerte (gerundet)

```
180°/π = 57,2958°   π/180° = 0,01745 rad/Grad
π = 3,14159         e = 2,71828
```

Text entlang Kurven (pst-text)

```
\pstextpath[justify]{path}{text}
```

Mögliche Werte für justify

```
l      c      r
```

Outline-Buchstaben (pst-char)

```
\pscharpath[par]{text}
\pscharclip[par]{text}...\endpscharclip
```

Wiederholungen (multido)

```
\multido{variables}{rep}{stuff}
```

Mögliche Variablentypen sind *Integer* (\i), *Dimension* (\d), *Number* (\n) und *Real* (\r). Ein Dekrement wird in der Form \nx=5.30+-1.25 angegeben.

Festkommaaddition und -subtraktion

```
\FPadd{num1}{num2}{command}
\FPsub{num1}{num2}{command}
```

Das Kommando *command* wird definiert und das Ergebnis darin abgespeichert.

Parameter für Farbgradienten (pst-grad)

```
fillstyle = gradient
gradbegin = color
gradend = color
gradlines = int
gradmidpoint = num
gradangle = ang
```

A Voreingestellte Werte

```
angle=0          hatchcolor=black
arcangle=8       hatchsep=4pt
arm=10pt         hatchwidth=0.8pt
arrowinset=0.4   labelsep=5pt
arrowlength=1.4  linearc=0pt
arrows=-         linecolor=black
arrowsize=2pt 3  linestyle=solid
border=0pt       linewidth=0.8pt
bordercolor=white loopsize=1cm
cornersize=relative ncurv=0.67
dash=5pt 3pt     nodesep=0pt
dimen=outer      nrot=0
dotangle=0       offset=0pt
dotscale=1       plotpoints=50
dotsep=3pt       plotstyle=line
dotsize=2pt 2    radius=2pt
dotstyle=*       runit=1cm
doublecolor=white shadow=false
doubleline=false shadowangle=-45
fillcolor=white  shadowcolor=darkgray
fillstyle=none   shadowsize=3pt
framearc=0       subgridcolor=gray
framesep=3pt     subgriddiv=5
gridcolor=black  subgriddots=0
griddots=0       subgridwidth=0.4pt
gridlabelcolor=black tbarsize=2pt 5
gridlabels=10pt  unit=1cm
gridwidth=0.8pt  xunit=1cm
hatchangle=45    yunit=1cm
```

Copyright © 2003–2005 by Uwe Siart <uwe@uwe-siart.de>

This material may be distributed only subject to the terms and conditions set forth in the *Open Publication License*, v1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

Diese Befehlsübersicht ist ein Auszug häufig verwendeter PSTricks-Befehle zur schnellen Referenz. Sie enthält *nicht* den gesamten Befehlsvorrat von PSTricks und ersetzt vor allem nicht die Lektüre der ausführlichen PSTricks-Dokumentation. Für weitere Informationen siehe <http://www.tug.org/applications/PSTricks/>, <http://www.pstricks.de> und

Voß, H.: *PSTricks*. Grafik mit PostScript für T_EX und L^AT_EX. 2. verb. Aufl. Berlin : Dante e. V. und Lehmanns Fachbuchhandlung, 2005

POSTSCRIPT® is a registered trademark of Adobe Systems Incorporated.