

R_X– ein Graphikpaket zur PostScript-Erzeugung

André Wobst

5. März 2004

DANTE 2004

Darmstadt

Inhalt

vormals: R_X := Postscript + Python + T_EX

heutzutage: R_X– python graphics package

- ▶ Motivation
- ▶ Kurze Einführung in Python
- ▶ Pfade als mathematische Objekte
- ▶ Ausgabe von Pfaden: Dekoratoren
- ▶ T_EX-Anbindung zur Erzeugung von Text
- ▶ Komplexe Graphikaufgaben (x-y-Graphen etc.)
- ▶ Entwicklung von R_X

Motivation

- ▶ programmierbares Graphiksystem
- ▶ Erzeugung frei skalierbarer Vektorgraphiken
- ▶ Umfangreiche elementare Graphikfähigkeiten
- ▶ Unterstützung für komplexe Graphikaufgaben
- ▶ Nutzung einer vorhandenen geeigneten Programmiersprache (für RX selbst und auch für die Benutzer)
- ▶ TEX für Textsatz (möglichst optimal integriert)
- ▶ Anderer Lösungen beachten

RX – ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Hello, world!

Beispiel hello.py:

```
from pyx import *  
c = canvas.canvas()  
c.text(0, 0, "Hello, world!")  
c.stroke(path.line(0, 0, 2, 0))  
c.writetofile("hello")
```

Ausführen von "python hello.py" ergibt "hello.eps":

Hello, world!

RX – ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Python 1

- ▶ Interpretierte Sprache (mit Byte-Code)
- ▶ Wenige Zeichen mit besonderer Bedeutung (gute Lesbarkeit)
- ▶ Streng typisiert, aber dynamisch zur Laufzeit
- ▶ Zahlen: `int`, `long` (beliebig lang), `float` (8 Byte), `complex`
- ▶ Zeichenketten: `s = "Hello, world!"` → `print s`, `s[0]`
- ▶ Listen: `l = [1, 2, 3]` → `print l[0]`, `l[1:2]`
- ▶ Dictionary: `d = {"a": 1, "b": 2}` → `print d["a"]`
- ▶ Programmblöcke durch Einrücken

*"We will perhaps eventually be writing only small modules which are identified by name as they are used to build larger ones, so that devices like **indentation**, rather than delimiters, might become feasible for expressing local structure in the source language."*

– Donald E. Knuth, "Structured Programming with goto Statements"
Computing Surveys **6**, 261 (1974)

RX– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Python 2

- ▶ **if-else-elif**-Konstrukt (Logik auch mit ganzen Zahlen):

```
if i or j:  
    print "i oder j ungleich Null"  
else:  
    print "i und j sind beide Null"
```
- ▶ **for**-Schleife (`range`-Funktion liefert Sequenz ganzer Zahlen):

```
for i in range(10):  
    print i
```
- ▶ **while**-Schleife (`len`-Funktion ermittelt Länge einer Sequenz):

```
while len(l):  
    print l[0]  
    l = l[1:]
```
- ▶ **for**- und **while**-Schleife unterstützen **continue**, **break** und **else**
- ▶ Fehlerbehandlung durch **try-except**-Blöcke

RX– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Python 3

- ▶ Funktionen:

```
def faculty(n):  
    if not n:  
        return 1  
    return n*faculty(n-1)
```

- ▶ Benannte und vorbelegte Argumente (ähnlich `keyval.sty`):

```
def f(a, b="default", c=[1, 2, 3]): ...
```

- ▶ Objekte:

```
class A: # inheritance via class B(A): ...  
    def show(self, a, b, c): ...
```

- ▶ spezielle Methoden: `__init__(self, ...)`, `__add__(self, other)`

- ▶ Instanzen:

```
a = A()  
a.show(1, 2, 3) # identisch zu A.show(a, 1, 2, 3)
```

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

R_X

- ▶ R_X steht unter GPL
- ▶ verfügbar unter <http://pyx.sourceforge.net/>, z.Z. Version 0.5.1
- ▶ Python-Paket (lauffähig ab Python 2.0)
Python verwendet PSF Lizenz (kein copyleft, GPL kompatibel)
Python ist verfügbar auf Vielzahl von Plattformen
- ▶ nutzt T_EX (oder L^AT_EX) zum Textsatz
basierend auf dvi-Ausgabe und type1-Fonts
- ▶ `from pyx import *` lädt unter anderem folgende Module:
path, trafo Pfade, lineare Transformationen
canvas, deco Zeichenfläche, Dekoratoren für Pfade
style, color Attribute beim Dekorieren von Pfaden
text T_EX-Anbindung
box, connector, graph Komplexere Graphikoperationen

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Koordinaten/Einheiten

- ▶ Ursprung ist links unten
- ▶ Standard-Einheit: 1 cm
- ▶ Kontext-abhängige Skalierungen:
 - ▶ Unskalierbar – t (für true)
 - ▶ Normale Koordinaten – u (für user)
 - ▶ Abstände, Symbolgrößen – v (für visual)
 - ▶ Liniendicken – w (für width)
 - ▶ (ggf. zukünftig x für T_EX)
- ▶ Rechteckige Umrandung (bounding box) automatisch (allerdings ohne Liniendicken etc.)
- ▶ Zentrierung der Ausgabe auf ein Papierformat und gleichzeitig verwendbar in L^AT_EX u.ä.

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Pfade

- ▶ mathematische Objekte (ohne Breite etc.)
- ▶ zusammengesetzt aus Pfadelementen
- ▶ Pfadelemente: moveto, lineto, curveto, arc, closepath, ...
- ▶ Erzeugung von Pfaden durch Klassen im Pfadmodul:

```
p = path(moveto(0, 0), lineto(1, 0))  
p.append(lineto(1, 1))  
p2 = line(0, 0, 1, 0) # auch rect, curve, circle
```
- ▶ Operationen auf (Norm-)Pfaden:
arclength Pfadlänge
lentopar Parametrisierung als Funktion der Länge
intersect Schnittpunkte zwischen Pfaden (Parametrisierung)
at Position auf Pfad (Parametrisierung)
split Pfadsegmente (Parametrisierung)
transformed transformierter Pfad

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

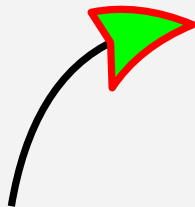
Dekoratoren

- ▶ verbinden Pfade und Zeichenattribute (Linienstile, Farben, etc.)
- ▶ können Pfade modifizieren und/oder Dinge hinzufügen
- ▶ canvas-Methode draw wendet Dekoratoren an:
`c.draw(p, [deco.stroked, deco.earrow.normal])`
- ▶ Linien- und Füll-Dekoratoren als canvas-Methoden:
`c.stroke(p, [deco.earrow.normal])`
`c.fill(path.rect(0, 0, 5, 5))`
- ▶ Zeichenattribute können direkt an draw, stroke- und fill oder aber an Dekoratoren übergeben werden:
`c.stroke(p, [color.rgb.blue, style.linestyle.dashed])`
`c.draw(p, [deco.filled([color.rgb.red]),
deco.stroked([color.rgb.green])])`

RX- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Beispiel: Pfeil

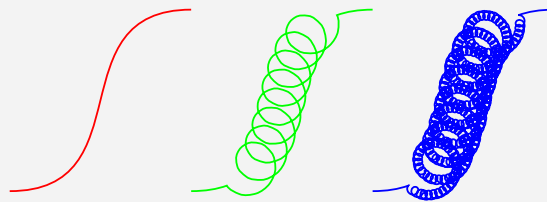
```
from pyx import *
c=canvas.canvas()
p=path.curve(0, 0, 0.05, 0.3, 0.2, 0.5, 0.5, 0.5)
a=deco.earrow.Large([deco.filled([color.rgb.green]),
deco.stroked([color.rgb.red, style.linejoin.round])])
c.stroke(p, [a])
c.writetofile("arrow")
```



RX- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Beispiel: Feder

```
from pyx06pre import *
c=canvas.canvas()
p = path.curve(0, 0, 1.5, 0, 0.5, 2, 2, 2)
c.stroke(p, [color.rgb.red])
c.stroke(p.transformed(trafo.translate(2, 0)),
        [color.rgb.green, deco.wriggle()])
c.stroke(p.transformed(trafo.translate(4, 0)),
        [color.rgb.blue, deco.wriggle(),
         deco.wriggle(loops=250, radius=0.05)])
c.writeEPSfile("wriggle")
```



R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Erzeugung von Text

- ▶ T_EX oder L^AT_EX wird als Prozess gestartet und überwacht (mehrere Instanzen möglich, aber normalerweise nicht nötig)
- ▶ zu verarbeitender Text wird in Box gesetzt; Boxgröße via stdout zurückgeben; Inhalt der Box als einzelne Seite in DVI Datei
- ▶ dvi lesen bei Bedarf (-ipc Option oder T_EX beenden)
- ▶ R_X erzeugt aus DVI Seiten canvas-Instanzen
- ▶ ausschliesslich Type1-Schriften (wie üblich psfonts.map etc.)
- ▶ Unterstützung virtueller Schriften
- ▶ Reduktion auf benutzte Glyphen (via C-extension module)
- ▶ Unterstützung wohldefinierter \special-Anweisungen; pyx.def für graphic[s/x] und color-Pakete
- ▶ Marker für Positionen im Text (via \special)

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

TeX-Details 1

```
from pyx import *
text.set(texdebug="text.tex")
c = canvas.canvas()
c.text(0, 0, "Hello, world!")
c.writetofile("text")
```

- ▶ text-Modul stellt Klasse texrunner sowie eine Instanz namens defaulttexrunner zur Verfügung
- ▶ text.set für Konstruktor-Einstellungen (L^AT_EX-Klasse etc.)
- ▶ Präambel-Modus: text.preamble(r"\usepackage{graphicx}")
- ▶ c.text(...) ist identisch zu:
c.insert(text.text(...)) # bzw. noch genauer
c.insert(text.defaulttexrunner.text(...))
- ▶ obiges Beispiel erzeugt lauffähige Datei text.tex

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobst
da

TeX-Details 2

```
This is TeX, Version 3.14159 (Web2C 7.4.5)
(./text.tex
! Undefined control sequence.
1.5 \raiseerror
      %
PyXInputMarker:executeid=1:
PyXInputMarker:executeid=2:
PyXInputMarker:executeid=3:
PyXInputMarker:executeid=4:
PyXBox:page=1,lt=0.0pt,rt=55.58344pt,ht=6.94444pt,dp=1.94444pt:
[80.121.88.1]
PyXInputMarker:executeid=5:
)
(see the transcript file for additional information)
Output written on text.dvi (1 page, 192 bytes).
Transcript written on text.log.
```

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobst
da

TEX-Details 3

```
% ...
\newbox\PyXBox%
\long\def\ProcessPyXBox#1#2{%
\setbox\PyXBox=\hbox{{#1}}%
\immediate\write16{PyXBox:page=#2,...,ht=\the\ht\PyXBox,...:}%
% ...
\ht\PyXBox0pt%
{\count0=80\count1=121\count2=88\count3=#2\shipout\box\PyXBox}}%
\def\PyXInput#1{\immediate\write16{PyXInputMarker:executeid=#1:}}%
\PyXInput{1}%
% ...
\ProcessPyXBox{Hello, world!%
}{1}%
\PyXInput{5}%
\end%
```

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

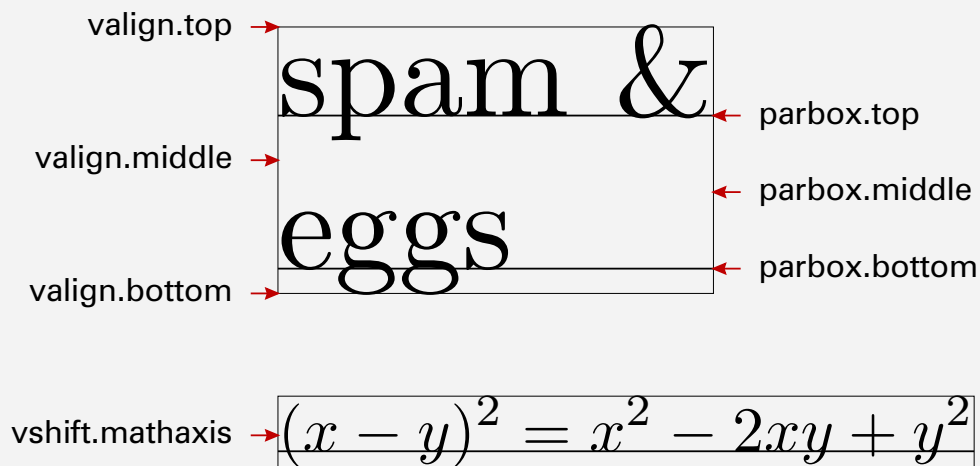
Textattribute

- ▶ horizontale Ausrichtung: `text.halign.[left|center|right]`
- ▶ vertikale Ausrichtung: `text.valign.[baseline|top|middle|bottom]`
- ▶ vertikale Box: `text.parbox(<Breite>)`; zusätzlicher Parameter `baseline` mit Varianten `top`, `middle` und `bottom`
- ▶ vertikal Verschieben mit `text.vshift` um Prozentsatz einer Texthöhe oder mathematische Achse (`text.vshift.mathaxis`)
- ▶ Schriftgröße `text.size.large` etc.
- ▶ Mathematikmodus mittels `text.mathmode`
- ▶ beliebige eigene TEX/L^AT_EX-Konstruktionen
- ▶ zusätzlich Füllattribute und Transformationen

R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

Beispiel: Textausrichtung

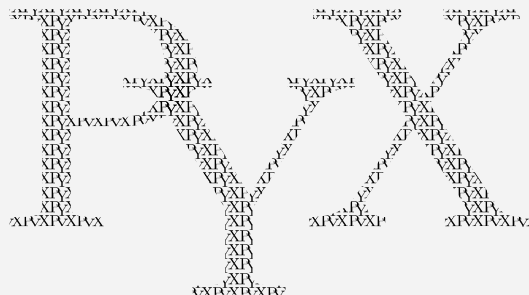


R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

Beispiel: Text mit Muster

```
from pyx import *  
p = canvas.pattern()  
p.text(0, 0, r"\PyX", [trafo.scale(0.5)])  
c = canvas.canvas()  
c.text(0, 0, r"\PyX", [trafo.scale(10), p])  
c.writetofile("pattern")
```

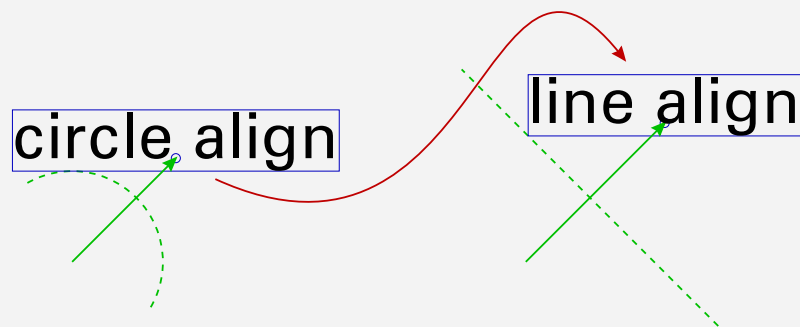


R_X– ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

Boxen und Konnektoren

- ▶ Boxen sind Zeichenflächen mit Berandung (derzeit nur konvexe Polygone; zukünftig Ausschneiden per Boxen, was momentan eine Zeichenflächeneigenschaft ist)
- ▶ Boxzentrum für Ausrichtung und Verbindung



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

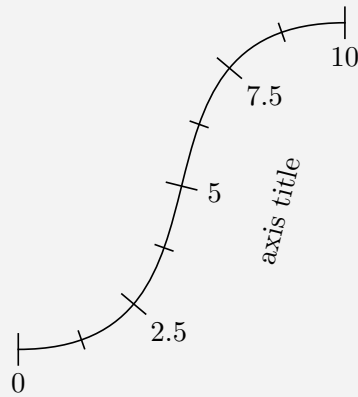
Achsen

- ▶ Elementarer Baustein für Graphen
- ▶ Automatische Achsenteilung unter Beachtung der Anzahl der Achsenstriche und Abstände (Ausdehnung) der Beschriftung
- ▶ Mischen manuell gesetzter Achsenstriche und automatischer Achsenteilung
- ▶ Achsenteilung mittels rationaler Zahlen
- ▶ Flexible Achsenbeschriftung
- ▶ Achsen mit Unterbrechung, Achsen für Balken-Graphen
- ▶ Entlang beliebiger Pfade
- ▶ erweiterbar auf echte Zeitachsen (experimentell)
- ▶ Quellen der folgenden Beispiele auf <http://pyx.sourceforge.net/>

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

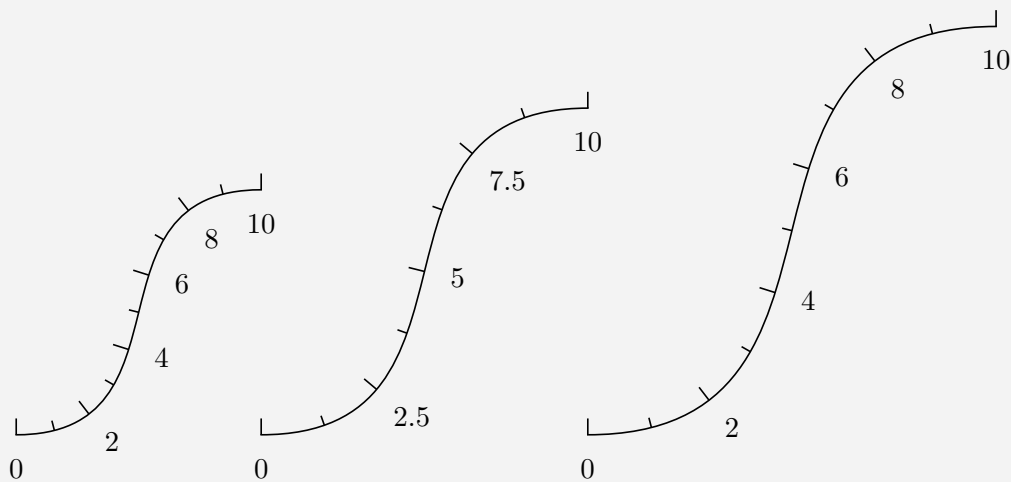
Beispiel: Pfad mit Achse



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

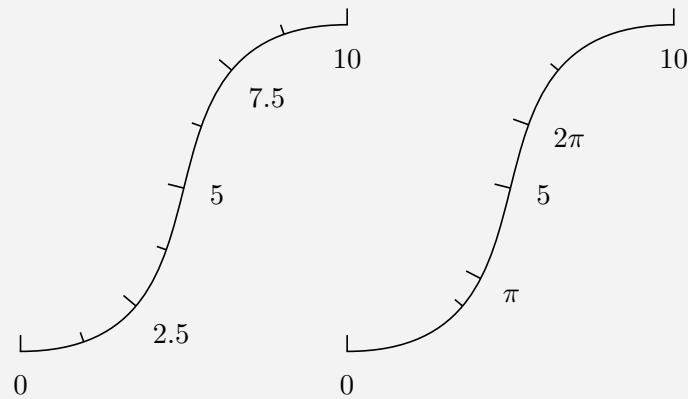
Beispiel: Achsenbewertung



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

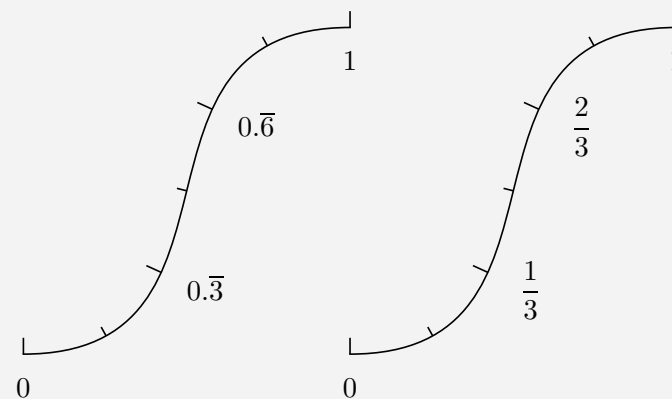
Beispiel: manuelle Achsenstriche



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

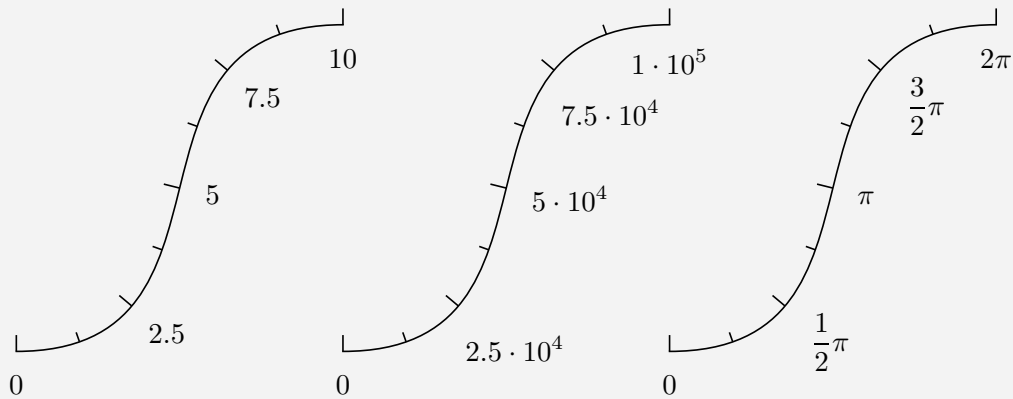
Beispiel: rationale Numerik



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta.de

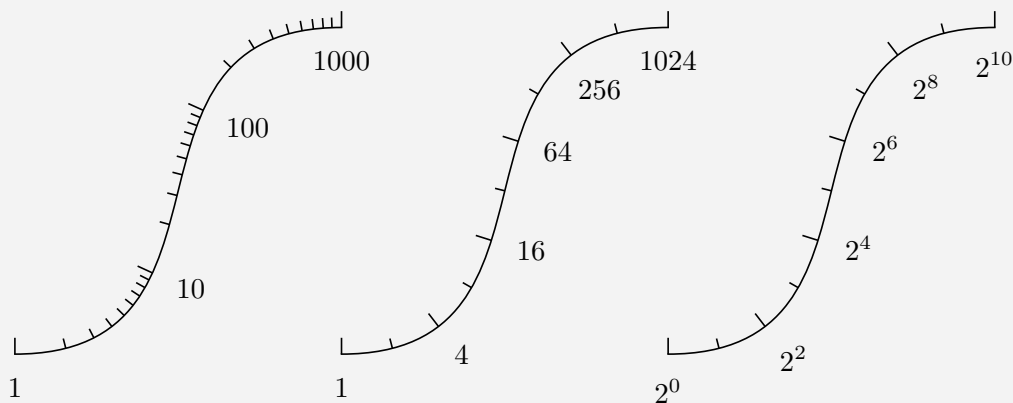
Beispiel: Achsenbeschriftung



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

Beispiel: logarithmische Achsen



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

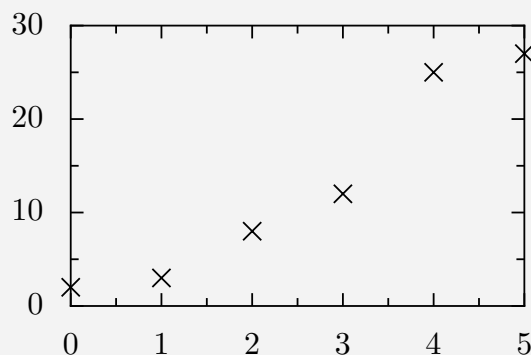
x-y-Graphen

- ▶ zusammengesetzt aus Komponenten, die meist auf beliebige Graphgeometrien anwendbar sind
- ▶ Beliebig viele Achsen; verknüpfbar auch über Graphen hinweg
- ▶ Daten aus Dateien, als Funktion oder Parameterfunktion
- ▶ Stile: Symbole, Linien, Texte, Pfeile mit Größe und Richtung, Fehlerbalken, farbige Rechtecke, Balken sowie verschiedene Kombinationen dieser Stile
- ▶ verschiedene Daten mit verschiedenen Stilen in einem Graph
- ▶ Daten-Legenden

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Beispiel: einfacher Graph

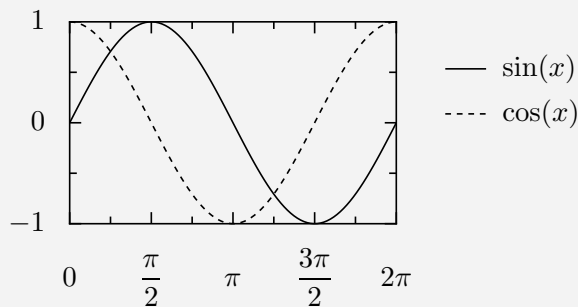
```
from pyx06pre import *  
g = graph.type.graphxy(width=5)  
g.plot(graph.data.data("minimal.dat", x=1, y=2))  
g.writeEPSfile("minimal")
```



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Beispiel: Funktionen

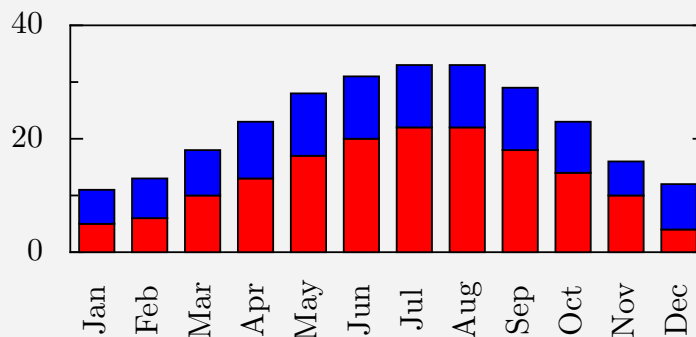
```
from math import pi
from pyx06pre import *
g=graph.type.graphxy(width=4, key=graph.key.key(hinside=0),
    x=graph.axis.linaxis(min=0, max=2*pi, divisor=pi,
        texter=graph.texter.rationaltexter(enumsuffix=r"\pi")))
g.plot(graph.data.function("y=sin(x)", title=r"\sin(x)"))
g.plot(graph.data.function("y=cos(x)", title=r"\cos(x)"))
g.writeEPSfile("piaxis")
```



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Beispiel: Balken

```
from pyx06pre import *
p=graph.painter.baraxispainter(nameattrs=[trafo.rotate(90)])
g=graph.type.graphxy(width=7, height=2.5,
    x=graph.axis.baraxis(painter=p))
g.plot(graph.data.data("bar.dat", xname=1, y=2, ystack=3),
    graph.style.bar())
g.writeEPSfile("bar")
```



R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

Entwicklung

- ▶ Gründung und Hauptentwickler: Jörg Lehmann, André Wobst
- ▶ Weitere Mitentwickler: Michael Schindler, Gert-Ludwig Ingold
- ▶ Projektstart mit CVS am 01.09.2000
- ▶ SourceForge.net Projekt seit 30.01.2002
- ▶ Erste Release 07.10.2002 (Version 0.1)
- ▶ Aktuell: Version 0.5.1 vom 22.01.2004
- ▶ derzeit und in naher Zukunft "alpha"-Software
- ▶ kommende Version 0.6 in (hoffentlich) einigen Wochen

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

Ziele

- ▶ Stabilisierung der Schnittstellen
- ▶ Entwicklung naheliegender Funktionalität:
z.B. Graphgeometrien, Boxen, Dekoratoren
- ▶ Entwicklung nicht so naheliegender Funktionalität:
z.B. Text in Boxen füllen (erste Experimente im CVS)
- ▶ zusätzliche Ausgabemedien:
PDF (erste Experimente im CVS) und SVG
- ▶ Dokumentation, Beispiele, Tests

R_X- ein Graphikpaket zur PostScript-Erzeugung
André Wobst, 5. März 2004, DANTE 2004, Darmstadt

wobsta
da

