

R_YX – eine Einführung

André Wobst

29. Oktober 2004

31 MV DANTE e.V.

Hannover

Inhalt

vormals: R_YX := Postscript + Python + T_EX

heutzutage: R_YX – Python graphics package

- ▶ Motivation
- ▶ „Hello, world!“
- ▶ Pfade als mathematische Objekte
- ▶ Ausgabe von Pfaden: Dekoratoren
- ▶ T_EX-Anbindung zur Erzeugung von Text
- ▶ Ausblick: Was R_YX noch so kann
- ▶ Entwicklung von R_YX

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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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.writeEPSfile("hello")
```

Ausführen von „python hello.py“ ergibt „hello.eps“:

Hello, world!

RX – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Python

- ▶ Interpretierte Sprache (mit Byte-Code)
- ▶ Wenige Zeichen mit besonderer Bedeutung (gute Lesbarkeit)
- ▶ Streng typisiert, aber dynamisch zur Laufzeit
- ▶ Datentypen: Zahlen (einschliesslich beliebig lange Ganzzahlen) Fließkommazahlen, Zeichenketten (Unicode), Listen, Dictionaries
- ▶ Objektorientiert (Pfade etc. in R_X alles Instanzen von Objekten)
- ▶ Programmstrukturen: Verzweigungen, Schleifen, Exceptions etc.
- ▶ 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)

R_X – eine Einführung

André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover



R_X

- ▶ R_X steht unter GPL
- ▶ verfügbar unter <http://pyx.sourceforge.net/>, z.Z. Version 0.7
- ▶ 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, affine Transformationen
canvas, deco Zeichenfläche, Dekoratoren für Pfade
style, color Attribute beim Dekorieren von Pfaden
text T_EX-Anbindung
- ▶ Weitere Module: deformer, box, connector, bitmap, graph, etc.

R_X – eine Einführung

André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover



Koordinaten/Einheiten

- ▶ Ursprung ist links unten
- ▶ Standard-Einheit: 1 cm
- ▶ unabhängige Skalierungen:
 - ▶ Unskalierbar – t (für „true“)
 - ▶ Normale Einheiten – u (für „user“)
 - ▶ Abstände, Symbolgrößen – v (für „visual“)
 - ▶ Liniendicken – w (für „width“)
 - ▶ Textgröße – x (für „TEX“)
- ▶ 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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Pfade

- ▶ mathematische Objekte (ohne Breite etc.)
- ▶ zusammengesetzt aus Pfadelementen
- ▶ orientiert sich an PostScript
- ▶ 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
```
- ▶ mehrere Teilpfade innerhalb eines Pfads (moveto oder „+“)
- ▶ Verbinden von Pfaden (join, joined oder „<<“)

R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Beispiel: Pfad

```
from pyx import *
c = canvas.canvas()
p1 = path.line(0, 0, 0.1, 0)
p2 = path.line(0.1, 0, 0.1, 0.1)
pa = p1 + p2
pb = p1 << p2
c.stroke(pa)
c.stroke(pb, [trafo.translate(0.2, 0)])
c.writeEPSfile("path")
```



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Normpfade

- ▶ bestehen nur aus geraden Linien und Bezier-Kurven
- ▶ jeder Pfad kann in Normpfad umgewandelt werden
- ▶ endliche Genauigkeit (minimale Länge von Pfadelementen)
- ▶ Operationen auf Normpfaden (ggf. automatische Umwandlung):
 - 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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Beispiel: Schnittpunkte 1

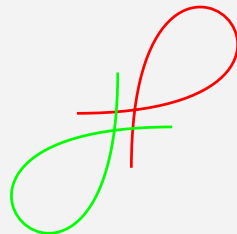
```
from pyx import *  
c = canvas.canvas()  
p1 = path.curve(-0.4, 0, 2, 0, 0, 2, 0, -0.4)  
c.stroke(p1)  
c.writeEPSfile("intersect1")
```



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Beispiel: Schnittpunkte 2

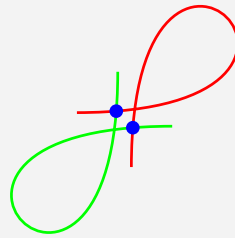
```
from pyx import *  
c = canvas.canvas()  
p1 = path.curve(-0.4, 0, 2, 0, 0, 2, 0, -0.4)  
p2 = p1.transformed(trafo.rotate(180, -0.05, -0.05))  
c.stroke(p1, [color.rgb.red])  
c.stroke(p2, [color.rgb.green])  
c.writeEPSfile("intersect2")
```



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Beispiel: Schnittpunkte 3

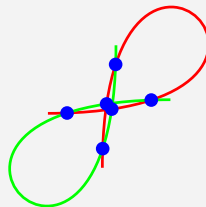
```
from pyx import *
c = canvas.canvas()
p1 = path.curve(-0.4, 0, 2, 0, 0, 2, 0, -0.4)
p2 = p1.transformed(trafo.rotate(180, -0.05, -0.05))
c.stroke(p1, [color.rgb.red])
c.stroke(p2, [color.rgb.green])
for t in p1.intersect(p2)[0]:
    x, y = p1.at(t)
    c.fill(path.circle(x, y, 0.05), [color.rgb.blue])
c.writeEPSfile("intersect3")
```



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Beispiel: Schnittpunkte 4

```
from pyx import *
c = canvas.canvas()
p1 = path.curve(-0.4, 0, 2, 0, 0, 2, 0, -0.4)
p2 = p1.transformed(trafo.rotate(180, 0.05, 0.05))
c.stroke(p1, [color.rgb.red])
c.stroke(p2, [color.rgb.green])
for t in p1.intersect(p2)[0]:
    x, y = p1.at(t)
    c.fill(path.circle(x, y, 0.05), [color.rgb.blue])
c.writeEPSfile("intersect4")
```



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

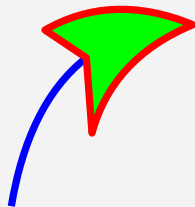
Dekoratoren

- ▶ verbinden Pfade und Zeichenattribute (Linienstile, Farben, etc.)
- ▶ können Pfadestücke ausblenden 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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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)
d = [deco.filled([color.rgb.green]),
     deco.stroked([color.rgb.red, style.linejoin.round])]
a = deco.earrow.Large(d)
c.stroke(p, [color.rgb.blue, a])
c.writeEPSfile("arrow")
```



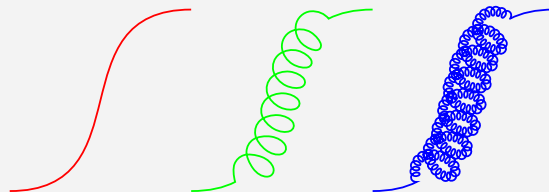
RX – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

Deformatoren

- ▶ modifizieren Pfade
- ▶ Transformationen sind unter anderem Deformatoren
- ▶ in der Regel nicht vertauschbar
- ▶ deform Methode
- ▶ bei draw, stroke und fill anwendbar
(werden zu Beginn nacheinander angewandt)

Beispiel: Feder

```
from pyx import *  
mycycloid = deformer.cycloid(skipfirst=0.5, skiplast=0.5)  
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, [trafo.translate(2, 0), color.rgb.green,  
            mycycloid(radius=0.2, halfloops=15)])  
c.stroke(p, [trafo.translate(4, 0), color.rgb.blue,  
            mycycloid(radius=0.2, halfloops=15),  
            mycycloid(halfloops=251, radius=0.05)])  
c.writeEPSfile("cycloid")
```



Erzeugung von Text

- ▶ $\text{T}_{\text{E}}\text{X}$ oder $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 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 $\text{T}_{\text{E}}\text{X}$ 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} – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

$\text{T}_{\text{E}}\text{X}$ -Details 1

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

- ▶ `text`-Modul stellt Klasse `texrunner` sowie eine Instanz namens `defaulttexrunner` zur Verfügung
- ▶ `text.set` für Konstruktor-Einstellungen ($\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -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} – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover



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 – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

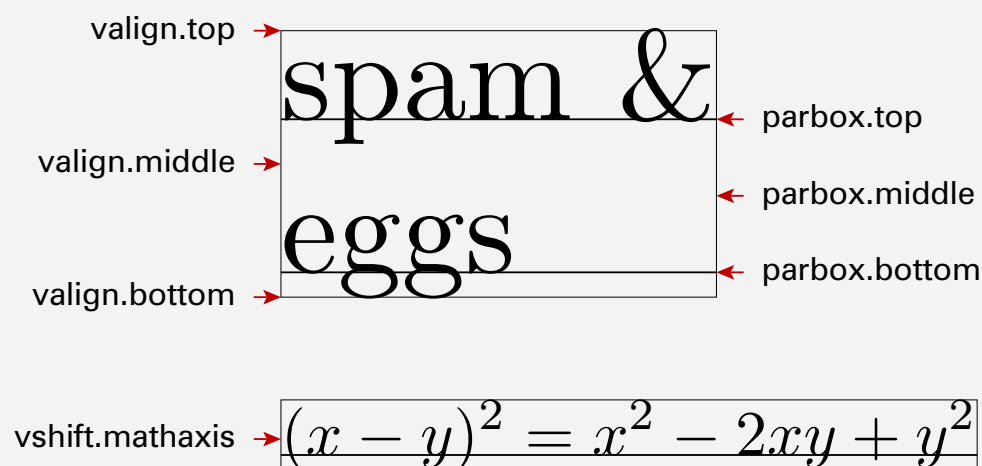


Textattribute

- ▶ 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 $\text{T}_\text{E}\text{X}/\text{L}\text{A}\text{T}_\text{E}\text{X}$ -Konstruktionen
- ▶ zusätzlich Füllattribute und Transformationen

R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

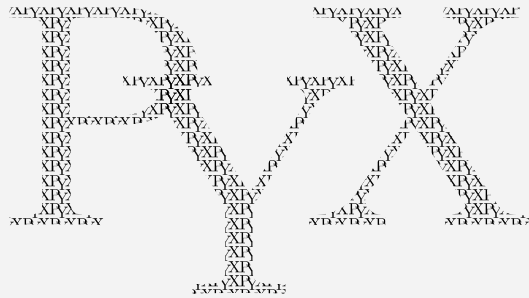
Beispiel: Textausrichtung



R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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.writeEPSfile("pattern")
```

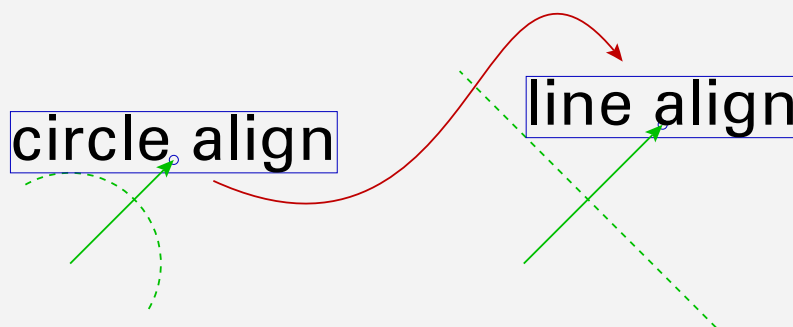


R_X – eine Einführung

André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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 – eine Einführung

André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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)
- ▶ Beispiele auf <http://pyx.sourceforge.net/>

Rx – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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 Kombinationen dieser Stile
- ▶ verschiedene Daten mit verschiedenen Stilen in einem Graph
- ▶ Daten-Legenden
- ▶ Beispiele auf <http://pyx.sourceforge.net/>

Rx – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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 0.7 vom 21.10.2004
- ▶ derzeit und in naher Zukunft "alpha"-Software

R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

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 (experimentell vorhanden)
- ▶ zusätzliche Ausgabemedien:
PDF (experimentell vorhanden) und SVG
- ▶ Dokumentation, Beispiele, Tests

R_X – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover

wobsta
da

Fragen

Fragen?

contact@wobsta.de
<http://www.wobsta.de/>

wobsta.de

RX – eine Einführung
André Wobst, 29. Oktober 2004, 31 MV DANTE e.V., Hannover