Importing Graphics for Statistical Plots

Paul Murrell

August 10 2006



◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Introduction

Old hat

• A statistical graphics system is most commonly used as only a **producer** of graphical images.



▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQで

Introduction

New hat

• It is also useful to make the statistical graphics system a **consumer** of graphical images.



Why import graphics?

• Adding company/institution logos



◆□ ▶ ◆□ ▶ ◆ □ ▶ ◆ □ ▶ ◆ □ ▶ ◆ ○ ◆ ○ ◆

Why import graphics?

• Backgrounds/watermarks



Why import graphics?

• Custom plotting symbols



Opening Gambits of Louis Charles Mahe De La Bourdonnais

Why import graphics?

• Chart Junk!



▲□▶ ▲□▶ ▲ 三▶ ▲ 三▶ 三 のへぐ

Importing graphics with R

- The **pixmap** package already provides facilities for importing bitmap images.
- We would like a mechanism for importing vector images.
 - vector images scale
 - for some images, the output will be much smaller than for a bitmap

▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

Solution Statement

Target only PostScript

- There are many (free) tools for converting to PostScript from other formats: ghostscript and xpdf (PDF), xfig (FIG), InkScape (SVG).
- PostScript is a sophisticated graphics language, so very complex images can be represented
- There is an open source interpreter for PostScript (ghostscript) so we do not have to write an interpreter
- PostScript is a programming language so we can write a PostScript program to export other PostScript images

◆□▶ ◆□▶ ◆□▶ ◆□▶ □ ○○○



Solution Statement

Convert to intermediate XML format

• XML is plain text, but with a discoverable structure, and a natural support for storing hierarchical information, etc ...

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQで

- XML can be read by R
- XML can be read by other software
- XML can be produced by other software
- XML can be converted to other formats (e.g., SVG)



Solution Statement

Read into general R object

- We may want to draw the picture using **grid** or using **traditional** graphics
- The image information is data; we may want to transform the image before drawing it, or we may want to just analyse the image
- These general R objects can be created from information other than imported PostScript files



The **grImport** package makes it possible to import external PostScript images for use within an R plot.



▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQで

```
%!PS-Adobe-2.0 EPSF-1.2
%%Creator: Adobe Illustrator(TM)
%%For: OpenWindows Version 2
%%Title: tiger.eps
. . .
.8 setgray
clippath fill
-110 -300 translate
1.1 dup scale
0 g
0 G
0 i
0 J
0 ј
0.172 w
10 M
[]0 d
0 0 0 0 k
. . .
```



. . .

```
PostScriptTrace("tiger.ps")
```

```
tiger <-
   readPicture("tiger.ps.xml")</pre>
```

```
pushViewport(plotViewport())
grid.picture(tiger)
```



▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG"
"http://www.w3.org/TR/2001/REC-SVG...">
<!-- Created with Sodipodi -->
<svg version="1.0">
...
<g
    style="font-size:12;"
    id="g874">
</path
    d="M 0 437 L 437 0 "
    style="fill:none;fill-opacity:1"
    id="path616" />
...
```

Convert SVG to PostScript
using InkScape

```
PostScriptTrace("chess.ps")
```

```
chess <-
readPicture("chess.ps.xml")</pre>
```



・ロト ・ 国 ト ・ ヨ ト ・ ヨ ト

The picturePaths() function draws individual paths from a picture, which makes it possible to identify elements of a picture.

"Picture" objects can be subsetted, which makes it possible to extract elements of a picture.

picturePaths(chess[125:136])



▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQで



Opening Gambits of Louis Charles Mahe De La Bourdonnais

pawn <chess[205:206]

	1931
Trebi	X XX XX X
Wisconsin No. 38	XXX X X
No. 457	IX II I
Glabron	X X X X
Peatland	IXI I I
Velvet	II X I
No. 475	III I II
Manchuria	XIIII
No. 462	XXX X X X
Svansota	XXXXX
	1932
Trebi	I X IX
Wisconsin No. 38	X X X X X
No. 457	XXX X X
Glabron	I I IX
Peatland	MII I
Velvet	IIIX
No. 475	I III II
Manchuria	XX XX
No. 462	III I II
Svansota	IXIII
	20 30 40 50 60
	< □ > < 🗗 >ield ≧ > < ≧ > 🦉

Issues

R graphics is primitive

- PostScript can do things that R graphics cannot
 - R only supports filling regions using the "winding rule" (PostScript also supports "even-odd" filling)
 - R does not support general paths (e.g., disjoint paths)
 - R does not support general clipping regions
- Other vector formats can do things that PostScript cannot
 - PDF and SVG support transparency, image composition operators, ...
 - Conversions may appear to work, but you might end up with a bitmap in your PostScript file
- It is not easy/possible to export all of every PostScript file
 - It is possible to export PostScript text as a path, but it is not ideal, and in most cases probably illegal
 - It is unclear how to export a bitmap from a PostScript image

A Picture object is data

The R object representing an external vector image can be treated like any other data source; it can be transformed, subsetted, augmented ... even **repaired**.

PRD <- readPicture("PRD_XUS_s_red.eps.xml")
grid.picture(PRD)</pre>





Issues



シック 単 (中本) (中本) (日)

Issues

Healing the logo

The single complex path can be processed to produce multiple disjoint paths.

newPRD <- explodePaths(PRD)
grid.picture(newPRD)</pre>





Healing the logo The new disjoint paths can be reordered.

```
grid.picture(newPRD)
grid.picture(newPRD[c(264, 268, 278:279, 284)], ...)
grid.picture(newPRD[180], ...)
```



Conclusions

Disclaimer

The grImport package is not a licence to violate copyright licences!

Summary

- It is useful to be able to import vector images for use in statistical plots.
- PostScript is a good external graphics format to target.
- An intermediate XML format allows alterative sources and destinations for graphics.
- An intermediate R object allows the picture to be treated as "just" data.
- There are limits to what R graphics can do, but with a bit of imagination there are some reasonable workarounds.

Acknowledgements

- Richard Walton made significant improvements to the grImport code last (Southern) Summer.
- The conversion of PostScript files is performed using ghostscript http://www.cs.wisc.edu/~ghost/
- The tiger image is part of the **ghostscript** distribution; the tiger data are from http://www.globaltiger.org/population.htm.
- The greyscale version of the tiger used the colorspace package by Ross Ihaka.

- The chess board image (by Jose Hevia) is from the Open Clip Art Library http://openclipart.org/clipart//recreation/games/chess/chess_game_01.svg
- The chess data are from chessgames.com http://www.chessgames.com/perl/chess.pl?page=1&pid=31596
- The man/woman picture is from the **Cisco Network Topology Icons** http://www.cisco.com/web/about/ac50/ac47/2.html
- The "problem logo" is used with kind permission from J&JPRD