



Postscript

Specialisation Seminar

Evin Aydin

Postscript

Introduction

Postscript

PostScript is a powerful and versatile programming language specifically designed for professional use in the printing and graphics industry, capable of creating high-quality, scalable vector graphics.

Postscript

History

History

PostScript revolutionized the computer industry by providing a solution for consistent high-quality printing and graphics rendering on diverse devices, establishing Adobe as a standard.

- **Inventors:** John Warnock, Charles Geschke
- **Problem:** Inconsistent font and graphics rendering
- **Solution:** Mathematical curves
- **Benefits:** Precise alignment, high-quality rendering
- **Impact:** Industry standard, transformation of printing and publishing

Postscript Language

Overview

- 1 **Generating Postscript files**
- 2 **Interpreter**
- 3 **Basic elements**

Postscript Language

Generating Postscript files

- 1 Manually Writing PostScript code using the PostScript language.
- 2 Utilizing system drivers or application programs that can generate PostScript files.
- 3 Using converters or filters to automatically convert other file formats into PostScript.

Postscript Language

Interpreter

Interpreter

PostScript, being an interpreted language, supports multiple interpreters that control output devices and offer different interaction modes, such as conventional printing, integrated display, and interactive programming, facilitating document interchange and management.

Interpreter

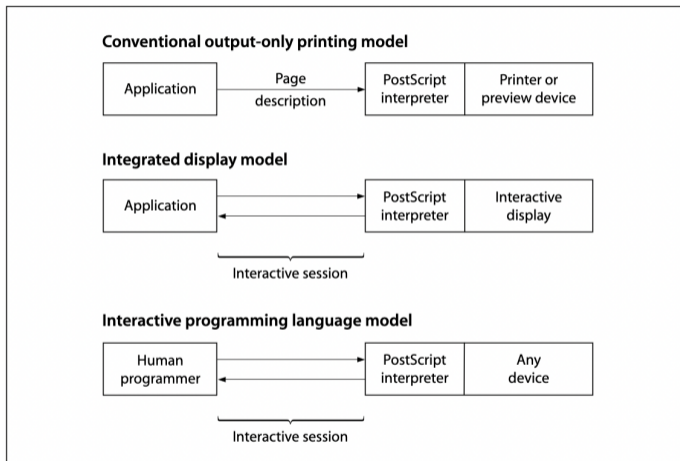


Figure: Postscript Interpreter application interact [1, Page 16]

Postscript Language

Basic elements

Basic elements

In PostScript, data is categorized into simple and composite types, where simple types are replicated when copied, including value, data type, and attributes, while composite types create references to the original values, resembling pointers in C.

Different data types

SIMPLE OBJECTS

boolean

fontID

integer

mark

name

null

operator

real

COMPOSITE OBJECTS

array

dictionary

file

gstate (*LanguageLevel 2*)

packedarray (*LanguageLevel 2*)

save

string

Figure: Different data types [1, Page 34]

Postscript Language

Stack

- ① Operand stack: operands and results
- ② Dictionary stack: name searches
- ③ Execution stack: intermediate execution stages
- ④ Graphics state stack: graphics parameters
- ⑤ Clipping path stack: current clipping path

Distinguished Features of Postscript

Operations on graphic elements

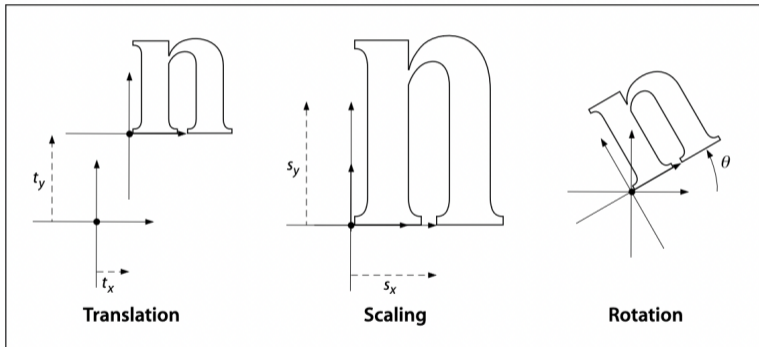


Figure: Different data types [2]

Distinguished Features of Postscript

Fonts

- ① Fonts stored as vectors for precise scaling, rotation, and transformation
- ② Wide selection of preinstalled and downloadable fonts
- ③ Support for grayscale and color fonts
- ④ Seamless integration of text and graphics
- ⑤ Encryption of font files for data protection

Distinguished Features of Postscript

EPS

- ① Independent of the output technology used
- ② Contains PostScript instructions for rendering graphics
- ③ Includes a low-resolution preview for non-PostScript screens
- ④ Supported by graphic design, desktop publishing, word processing programs
- ⑤ Exported from graphic design programs or converted from other formats
- ⑥ Specific instructions for scaling, moving, or cropping the graphic

Distinguished Features of Postscript

PDF

- ① Developed as an extension to PostScript
- ② Platform-independent file format for digital documents
- ③ Enhanced features for screen display and interaction
- ④ Supports hyperlinks, forms, multimedia, and interactivity
- ⑤ Consistent display across different systems and devices
- ⑥ PDF is a standalone file format, unlike PostScript
- ⑦ PDF files can be opened without a PostScript interpreter
- ⑧ Adobe Acrobat software for viewing, editing, and converting

Distinguished Features of Postscript SVG

- ① is an XML-based vector image format.
- ② Widely used for web graphics, supported by modern browsers
- ③ Resolution-independent, scalable without quality loss
- ④ Supports shapes, text, colors, gradients, and transparency
- ⑤ Interactivity can be added with scripting languages
- ⑥ for versatile , visually appealing design

Limitations of Postscript

Transparency Problem

- 1 PostScript transparency: Limited to 1-bit opacity
- 2 Image masks: Used for pixel-level control
- 3 No true transparency: Only show/hide image portions
- 4 Flattening process: Divide overlapping objects into smaller parts

Limitations of Postscript

Implementation Problem

Two types of limits architectural limits and memory limits:

Architectural limits

The limitations in the architecture of PostScript result from hardware-imposed constraints, such as the 32-bit limit on integers, and software design choices.

Memory limits

PostScript memory limits determine the maximum amount of available memory for the interpreter, affecting the handling of memory-intensive objects.

32

Postscript

Postscript compared to LaTeX

- *PostScript*
 - ① Page description language
 - ② Graphics manipulation
 - ③ Vector-based graphics
- *LaTeX*
 - ① Document preparation system
 - ② Text-focused
 - ③ Structural markup
 - ④ Mathematical typesetting

Postscript

Postscript compared to LaTeX

Combining Postscript and LaTeX

PostScript font packages in LaTeX enable the seamless incorporation of PostScript fonts into LaTeX documents, enhancing the typography and visual appearance.

- *Useful packages*

- ① Helvet
- ② Mathpazo
- ③ Mathptmx
- ④ Mathppl

Postscript

Conclusion

Conclusion

PostScript is a versatile and powerful programming language with a wide range of functions and operators, enabling complex graphic effects and seamless integration of text and graphics. It remains relevant in the printing and graphic industry, serving as the foundation for important formats like PDF and playing a crucial role in generating high-quality, scalable graphics.