Session
AD004

PostScript Programming

Robert Gezelter Software Consultant
35 – 20 167th Street, Suite 215
Flushing, New York 11358 – 1731
United States of America

+1 718 463 1079
gezelter@rlgsc.com

Monday, December 4, 1995
5:00 pm – 5:50 pm
Room 104

Fall 1995 US DECUS Symposium
Moscone Convention Center
San Francisco, California

Robert Gezelter +1 718 463 1079
Software Consultant 35 – 20 167th Street, Suite 215, Flushing, New York 11358 – 1731 USA

PostScript is a device independent Page Description Language with general programming capabilities.

Robert Gezelter
Software Consultant

References:

- ***PostScript Language series by Adobe***
  - ***Reference Manual – Red***
  - ***Tutorial and Cookbook – Blue***
  - ***Program Design – Green***

Robert Gezelter
Software Consultant

NOTES

References (cont'd):

- ***other authors***

Holzgang, David
"Understanding PostScript " Programming"
Sybex

Broswell, Frank
"Inside PostScript"
Peachpit Press

Robert Gezelter
Software Consultant

References (cont'd):

- ***other authors***

Holzgang, David
"PostScript Programmer's Reference Guide"
Scott Foresman & Co

Thomas, Barry
"A PostScript Cookbook"
Van Nostrand Reinhold

Robert Gezelter
Software Consultant

NOTES

References (cont'd):

- ***other authors***

Roth, Stephen
"Real World PostScript"
Addison Wesley

Glover, Gary
"Running PostScript
from MS-DOS"
Windcrest Books

Robert Gezelter
Software Consultant

There are two classes of concepts in PostScript:

- ***the imaging model; and***
- ***the programming model***

Robert Gezelter
Software Consultant

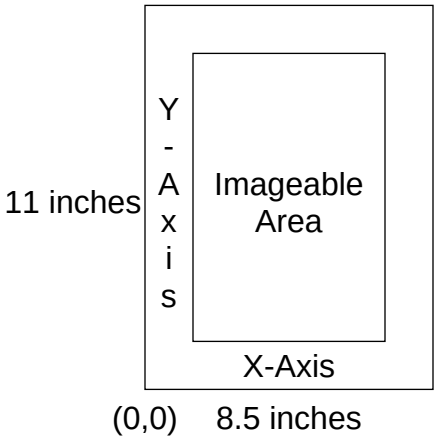
NOTES

The PostScript Imaging Model

- ***Coordinate system***
- ***Paints (a.k.a. inks)***
- ***Paths***

Robert Gezelter
Software Consultant

Coordinate System



- ***All positions are in terms of x,y coordinates***
- ***Initial Units Points (72 points ≈ 1 inch)***
- ***Scalable***

Robert Gezelter
Software Consultant

NOTES

Paints

- ***PostScript uses a "Paint" to fill a path or shape***
- ***Paints can be any shade of Black/White/Gray (monochrome devices) or color (on color devices)***
- ***Intermediate shades are produced by halftoning***
- ***Paints are opaque***

Robert Gezelter
Software Consultant

Paths

You use PostScript commands to create "Paths" which are filled with "Paint"

```
newpath
1 inch 1 inch moveto
0 2 inch rlineto
1 inch 0 rlineto
1 inch -2 inch rlineto
closepath
fill
```

Robert Gezelter
Software Consultant

NOTES

Programming Model

- ***Stack based***
- ***Objects***
- ***Fonts***

Robert Gezelter
Software Consultant

Programming Model

- ***Stack based language***
- ***Operands pushed on stack***
- ***Operators operate on stack***

Robert Gezelter
Software Consultant

NOTES

Stack based language

			mul		
add			2315		6945
5	⇒	9	3	⇒	(string)
4			(string)		739
			739		

Robert Gezelter
Software Consultant

Objects

- ***numbers***
  5 -7 8.652
- ***strings***
  (abc) (A)
- ***fonts***
  /Helvetica findfont

Robert Gezelter
Software Consultant

NOTES

Fonts

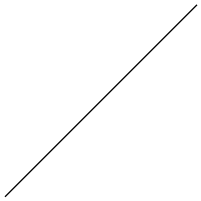
- ***collection of characters (can be filled, outlined, or stroked).***
- ***Are painted using the "show" command***

Robert Gezelter
Software Consultant

Example of PostScript

Draw a line.

```
newpath
0 0 moveto
2 inch 2 inch rlineto
stroke
```



Robert Gezelter
Software Consultant

NOTES

Figure Drawing

Draw a trapezoid!

```
newpath
1 inch 1 inch moveto
0 2 inch rlineto
1 inch 0 rlineto
1 inch -2 inch rlineto
closepath
fill
```


Robert Gezelter
Software Consultant

PostScript Text Example

Draw some 2 inch high characters in Helvetica

This is a test!

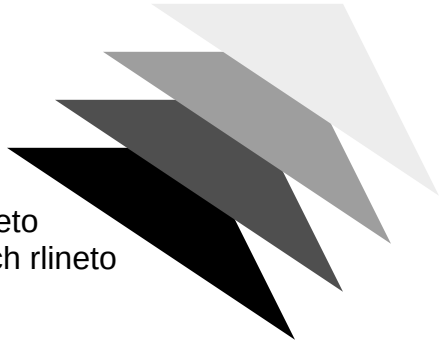
```
/Helvetica findfont
2 inch scalefont
setfont
(This is a test!) show
```


Robert Gezelter
Software Consultant

NOTES

Gray Scale and Opaqing Example

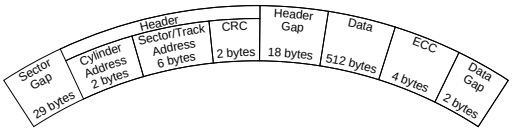
This example shows translation, loops, and opaquing.

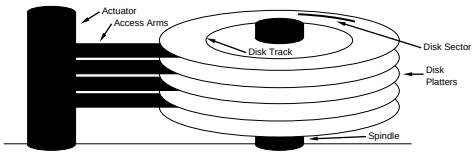


```
4 {
    newpath
    0 0 moveto
    2 inch 0 rlineto
    1 inch -2 inch rlineto
    closepath
    fill
    currentgray .31 add setgray
    .5 inch .5 inch translate
    } repeat
```


Robert Gezelter
Software Consultant

PostScript Power





On an SMD interface drive configured for use on a Digital system, 575 bytes are needed to store a 512-Byte data record. The 63-Byte overhead required to store each record also decreases the device's effective transfer rate.

This diagram was done as a PostScript programming problem

Robert Gezelter
Software Consultant

NOTES

Debugging PostScript Programs

- ***make use of interactive mode AND/OR exception handlers***
- ***work incrementaly***

Robert Gezelter
Software Consultant

Program Design Suggestions

- ***conform to the EPS standard***
- ***keep things structured***
- ***keep things manageable***
- ***watch out for side effects***

Robert Gezelter
Software Consultant

NOTES

PostScript is extremely powerful!

PostScript is also extremely seductive, excercise caution!

PostScript Programming
Slide 24

Questions?

Robert Gezelter Software Consultant
35 – 20 167th Street, Suite 215
Flushing, New York 11358 – 1731
United States of America

+1 718 463 1079
gezelter@rlgsc.com

PostScript Programming
Slide 25

NOTES