

Full Transparency From Raw PostScript

Don Lancaster

Synergetics, Box 809, Thatcher, AZ 85552

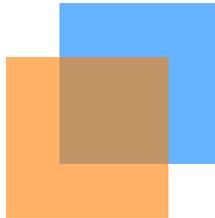
copyright c2002 as GuruGram #52

<http://www.tinaja.com>

don@tinaja.com

(928) 428-4073

The **PostScript** general purpose computing language initially used an **opaque** imaging model in which newer objects **completely covered** up any underlying material. **Transparency Effects** from raw **PostScript** in which a new object only **partially covers** (or otherwise cleverly manipulates) anything underlying that has gone before had been rather **cumbersome and ungainly** to date...



The latest (7.0 and higher) versions of **Acrobat .PDF** now include an updated version of **Acrobat Distiller 7.0**. Here you will find a brand new and sorely needed **/SetTransparency PDFMark Operator**. An operator that at long last makes doing transparency effects in raw **PostScript** fast and easy. Here is the raw **PostScript** code for our "two squares" example above...

```
<< /AllowTransparency true >> % enable transparency
setdistillerparams
  gsave 100 100 translate      % set position on page
    0.4 0.7 1 setrgbcolor     % draw a pastel blue box
    40 40 60 60 rectfill
  [ /ca .6 /SetTransparency   % set the transparency
    pdfmark
    1 0.5 0 setrgbcolor      % draw transparent orange
    20 20 60 60 rectfill
  grestore showpage
```

A copy of this demo [appears here](#). Note that this code **must** be sent to Distiller 7 or higher, but that the job options can be set as low as Acrobat 5. Note also that **transparency is normally disabled** in Distiller 7. And, when desired, must be activated by an `<</AllowTransparency true>> setdistillerparams`.

As of this revision, there appears to be a bug in Acrobat 7 that causes a page with transparency to appear "harsher" than a normal page **in the Acrobat viewer**. Some colors may also be rendered "wrong". You may note this here as a difference in viewability between page one and page two.

A possible cause is the 7.x reader currently and incorrectly forcing **CMYK** color on any transparency activity. This harshness shows up **on the reader only**; printing should be normal as expected. An **Adobe** fix during an upgrade can reasonably be expected. As a temporary workaround, placing transparency on **all** pages would cause the bug to be somewhat less obvious.

What is really neat about all this, is that there are at least **sixteen** different types of "transparency" available! Some of which do some really exotic things like dodges and burns and such. Here's a summary of the BM blend mode names...

| | | | |
|-------------------|-------------------|-------------------|-------------------|
| Color | Difference | Lighten | Overlay |
| ColorBurn | Exclusion | Luminosity | Saturation |
| ColorDodge | HardLight | Multiply | SoftLight |
| Darken | Hue | Normal | Screen |

More on these in the [PDF Reference Manual](#).

For More Help

Much more on these topics appear on our [PostScript](#) and [Acrobat](#) library pages. Free [Gonzo Utilities](#) and many use examples are found [here](#).

Additional consulting services are available per our [Infopack](#) services and on a contract or an hourly basis. Additional [GuruGrams](#) are found [here](#).

Further [GuruGrams](#) await your ongoing support as a [Synergetics Partner](#).