

# Book Cover Layouts Using my Gonzo Utilities

**Don Lancaster**

**Synergetics, Box 809, Thatcher, AZ 85552**

**copyright c2005 as [GuruGram #61](#).**

**<http://www.tinaja.com>**

**[don@tinaja.com](mailto:don@tinaja.com)**

**(928) 428-4073**

I have long had available a free set of my [Gonzo Utilities](#) that can be found in our [PostScript Library](#). Which I continue to aggressively personally use for **all** of my writing projects, my [PostScript as Language](#) programming research, and for countless other tasks.

These highly device independent and **proudly non-WYSIWYG PostScript** utilities can greatly simplify [Using PostScript as a General Purpose Computing Language](#), can offer **superbly fancy typesetting**, do **fancy modeling calculations**, very much **enhance gridding and graphing**, and even provide you with **unmatched quality electronic schematics**.

They also let you **read or write most any disk file in most any language**, thus leaping tall buildings in a single bound. Expanded, they may also be used to programmatically **write complex programs in other languages**. Such as those [JavaScript](#) Fourier Analysis routines found on our [Magic Sinewave](#) page. Or the incredibly legible small fonts in our [Bitmap Typewriter](#).

You can start learning the [Gonzo Utilities](#) with our [PostScript Beginner Projects](#). Or else go to most any [.PSL](#) file in our [GuruGram library](#) or elsewhere for highly detailed use examples.

To use the [Gonzo Utilities](#) yourself, you **download them** into a known place in your computer or place them on a floppy or other media. Then, **after making obvious site changes**, you include **one of the following** in the beginning of your standard ASCII textfile that will hold the [PostScript](#) instructions that you are going to send to [Acrobat Distiller](#)...

```
(C:\\Documents and Settings\\don\\Desktop\\  
gonzo\\gonzo.ps) run % use internal gonzo  
(A:\\gonzo.ps) run % use external gonzo
```

Be sure to note that...

**You MUST use a DOUBLE reverse slash inside a PostScript string every time you really want a single reverse slash!**

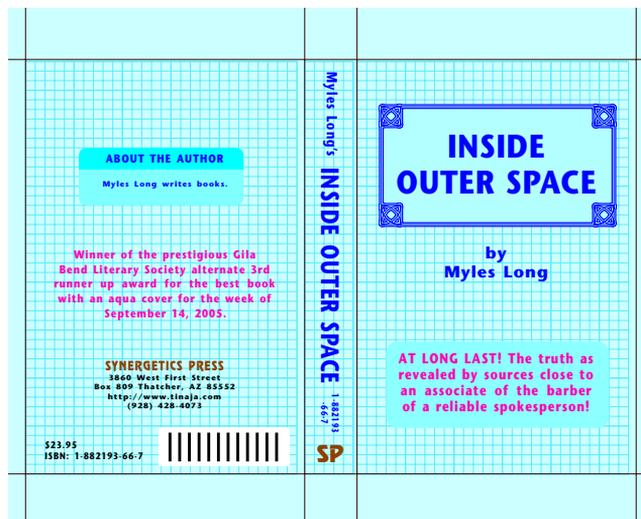
The **Gonzo Utilities** are normally in a group of activatable dictionaries. At present, their **activation sequence** of **gonzo begin ps.util.1 begin nuisance begin** is built into the last lines of the **Gonzo** download. Thus, Gonzo will automatically activate most of itself on its **run** command. The separate **electronics begin** dictionary is only sometimes needed, so you will have to activate it when desired.

In normal use, you will create an **ordinary ASCII textfile** using your favorite editor or word processor that has the (**...Gonzo...**) **run** command near its beginning. Typically, you will have a **showpage** at the end of your document. Note that you must **NOT** end with a **quit** or a **[D]** Control-D command!

You'll then send this file to **Acrobat Distiller**, which is included in all full featured versions of **Adobe Acrobat**. While Acrobat 7 is recommended at this writing, earlier versions can also be used. It is also sometimes possible to use **GhostScript** as an **Acrobat** replacement. My personal recommendation is to always use a genuine **Adobe** distiller instead.

## Gonzo Book Cover Layouts

There are just enough hassles and gotchas in doing book cover layouts for **BOD printing** that the **Gonzo Utilities** can be a good choice for this task. A typical book cover with design aides in place might look something like this...



I have put together a new set of utility routines as **BOOKCOVR.PSL** that should let you conveniently do most any book cover layout. A companion **BOOKCOVR.PDF** demo gives you a full size version of our above figure.

**BOOKCOVR.PSL** is fairly well documented internally, so what I thought I'd do here is just go over a few of the more subtle points of book cover layout. What follows is best understood if you do have copies of **BOOKCOVR.PSL** and **BOOKCOVR.PDF** printed out or in open windows.

The basic elements of **BOOKCOVR.PSL** are an oversize background color onto which three separate areas are placed for the front cover, the spine, and the rear cover. Each of these has their own internal coordinate system and is independent of the others.

Several design aides are included. **Region lines** outline the cover areas and can be handy during early layout, but should be made invisible otherwise. These are one or more sets of **crop marks** for pre- and post-binding cutting, and three separate **layout grids** for the cover art areas. These can be turned on or off by switching their commands from **true** to **false**. Such as **/showregions true store**.

## How Big?

One of the ruder surprises of book cover layout is that **most covers will demand an oversize printer**. A 5 x 8 inch paperback cover is about the largest you can hope to print on a conventional 8-1/2 x 11 inch printer. And then only if the spine is not too thick. Our **BOOKCOVR.PSL** allows any sane size cover layout.

Note that an ordinary printer with a "legal size" tray capability should be able to handle book covers up to around eight inches high by six inches wide with a one inch spine.

However...

**Larger book covers may require an oversize printer. If so, custom settings WILL be needed for Acrobat Distiller.**

**Go into EDIT ADOBE PDF SETTINGS and adjust the width and the height of the DEFAULT PAGE SIZE. Then resave with a new default settings filename.**

The example presently in **BOOKCOVR.PSL** is for a four by six inch book and will easily fit a conventional paper page and printer. For best printer compatibility, **the actual layout will physically remain in its portrait orientation**.

Unfortunately, Adobe Distiller has a disconcerting habit of deciding whether portrait or landscape is best for you. This is best defeated by using...

```
/currentdistillerparams where {pop}  
{userdict /currentdistillerparams {1 dict} put} ifelse  
/setdistillerparams where {pop}  
{userdict /setdistillerparams {pop} put} ifelse  
<</AutoRotatePages /None >> setdistillerparams
```

## Background Tasks

The positioning on the sheet is set by **xoffset and yoffset** as measured from the **lower left** of your portrait oriented sheet.

In general, the best covers will be done on special **duplex cover stock** that is **prefinished** with one color on one side and white on the other. (Note: This is a different use of the word "duplex" than you might be familiar with.) You can fake special stocks by simply printing a background color (or pattern or image) over a **background area** that is large enough to include needed trimming.

**Crop marks** can be placed on the background area to indicate the final trim size of the cover. A second set of crop marks may be desired if there is both a pre- and post-binding trim needed. As before, these can be turned on or off with a **/showcropmarks true store**. Further, **regions** can be placed around the front, spine, and rear covers. These are handy for layout, but should be removed later using a **/showregions true store**.

A "safety valve" is also present just in case an error in one of your page rotations or a text positioning value inadvertently moves you so far off the page that you cannot find it. To activate this feature, you simply uncomment the **% 300 400 translate 0.15 dup scale** debugging tool line.

## Front Cover

Any mix of text, line art, images, patterns, or area procs can be used on your cover layout. I prefer to work on a 10X grid as this makes most of the numbers short and convenient. The 0,0 orientation is at the **lower left**.

Your choice of fonts is particularly critical for a title, as there will be very few characters to properly absorb the font "vibes". I particularly like the variants on the **Stone** font. Be sure to **include** your fonts in your distillation. Otherwise, Distiller may **substitute** a multiple master font for you that may not look quite as good. Best results on the demo will happen with **Stone** and **Revue** available.

Custom kerning will often be needed to best show larger typography. In particular, the "A" in "SPACE" has two points of crowding towards the "P" and one point towards the "C". Negative kerning in Gonzo is done with a **bar-k**, while positive uses a **bar-j**.

One or more **centering variables** can be created as convenience operators. Such as **fcen**. This can **auto-track** for you should you make modest changes in the cover width. Auto-tracking cover height can be done but probably will create more problems than it solves.

An obvious caution...

**Any artwork that UNDERLAYS text or whatever MUST be called BEFORE the overlaying text is put down.**

The rounded box underlying the blurb was done using a stock **roundbox** routine in the **Gonzo Utilities**. That fancy border under the title was done using a sneaky and disgustingly elegant combination of **arcto crowding** and **superstroking**.

Details on this scheme appear in **Column 45** of our **PostScript Secrets**.

## The Spine

A key point before we continue...

**The spine width is CRUCIALLY SENSITIVE to the page count and the bulk and weight of the pages used.**

You will have to **very carefully measure** your needed spine width using the actual page count and actual papers selected for your final book production. While minor variations can sometimes be absorbed by changing your folding, this can and will uncenter your front and rear cover art if excessive.

A typical spine has a mix of normal and rotated text, so I decided to leave it in portrait mode with its 0,0 coordinate at **lower left**. As with the front and rear covers, the spine has its own grid system that you can turn on or off for layout. The **SP** service mark would appear horizontally. This could be an .eps artwork cut, a custom proc, or just the two-character **Revue** typography shown.

The three remaining text areas of the author, title, and ISBN are text separately rotated in place as needed. Again, large text custom kerning should be used where it improves appearance.

## Rear Cover and those ISBN Barcodes

The rear cover pretty much uses the same techniques as the front. A modified double **roundbox** can give you that two-tone **about the author** effect shown. The second **roundbox** clips the upper area to the proper size. Once again, be sure to put the background down first and the foreground last.

A crucial rule on book sales...

**If you want to sell your book on Amazon or in a bookstore or a regular store, you ABSOLUTELY MUST have an ISBN number for that book!**

**Your book also ABSOLUTELY MUST include a bar code in the LOWER RIGHT CORNER of your rear cover!**

ISBN numbers are normally gotten from the **US ISBN Agency**, which is currently administered by **Bowker**. Typical fees vary from three cents to three dollars each, with a ten ISBN number minimum order.

Your barcode can simply be pasted into our Gonzo layout proc. It is extremely important to **use the ISBN style barcode** and not one of the many other variants. Typically, a barcode will be an **.EPS** routine that you can either define inline inside the Gonzo layout or else **run** as an external file.

I've just used a fake barcode in our example layout. You can get a real barcode image from any of a number of free, shareware, or commercial programs. A barcode tutorial **appears here**. A free barcode generator **appears here**. One of many online interactive barcode generators **appears here**.

## Images

Images are easily included in your cover layout. Doing everything from a full image background to an author photo to an artwork enhancement. I have not included images in the demo in interest of keeping things fairly simple.

The best way to deal with an image is to make it into a **.EPS file**. Which can either be internally included as a **PostScript** proc or else externally **run** the same way we loaded in the **Gonzo Utilities**. Most other image formats are easily converted to .EPS format using the "**save as**" feature in **Adobe Illustrator** or a similar image management program.

Should all else fail, there is a special **DCTDecode** filter built into **PostScript** that you can use to directly read an accessible .JPG file and convert it into a PostScript image proc. Details on this in **jpg2pdf.pdf** in our **GuruGram** library.

## For More Help

Bunches of more info on Gonzo and PostScript in general can be found over on our **PostScript** library page. Along with lots of tutorials and utilities.

Additional consulting services are available per our **Infopack** services on a contract, seminar, or hourly basis. Further **GuruGrams** await your ongoing support as a **Synergetics Partner**.