



In This Issue	
Web Exclusives	Þ
Special Features	Þ
Free Subscriptions	Þ
Conferences/Events	
Editorial Info	Þ
Advertising Info	Þ
Reader Service	
Contact Us	
Partner Services	Þ



Have you changed any of your accounting practices in light of recent events?

> Yes No Vote

View Results Only



Dovebid Equipment Auctions Try our new used equipment auctions! Powered by DoveBid





CASSAVE THIS COMEMAIL THIS CASPRINT THIS PENNWELL

## PS, I love you

While some see PDF as the answer to all our prepress headaches, others are taking a more careful approach.

by Frank Merritt Braswell

The reality of prepress file processing today is that PostScript language files still dominate the scene and they aren't going away soon. For me, they're not taking away my PostScript files until they pry my cold dead fingers off the RIP. Unfortunately the RIPs in many shops are covered with cold dead fingerprints.



Don't get me wrong, I love and use Adobe Acrobat technology, but it won't lead to prepress heaven. The appropriate application of the Adobe PostScript and Acrobat technologies is the key to success. The problem is that as technology advances we must shift to adapt to better ways to solve our problems.

Frank Romano gave a talk several years ago at GATF's TechAlert

Conference where he covered some of the pitfalls of using portable document format (PDF) files for prepress. My conclusion was that there are about 1000 things that can go wrong with PDF file generation. Since there are about 1000 things that can go wrong with PostScript file generation, can PDF files be better?

A popular PDF website proclaims: "It is a fact that if you set up QuarkXPress properly to make a PostScript file, and if you set up Adobe Acrobat Distiller properly, you can make a PDF file that can be reliably used in high-end prepress and printing production environments." Notice the two critical "if you set ups" in that statement. Think about how many settings and options are involved in both of those setups. Not only that, but the original PostScript file generated by QuarkXPress contains all the information necessary for high-end prepress processing already.

I will examine some of the core problems and propose some solutions. If you've guessed that this is about preflight concepts, you're right. In many ways the problems we faced in years past are the same ones we face today. Problems with fonts, images, color, and PostScript errors are still with us.

## The real issue

The truth is that we must be sure that the prepress files we generate meet our print requirements regardless of what print file format we use. The transfer of files from the computer where they are generated to the computer where they are to be imaged is where the problems arise. If we could be certain that all the elements necessary for printing were present in the print file, imaging would almost always be successful.

It is important to realize that application files do not create the final image, PostScript files do. Just because the page looks good on your screen doesn't mean it will print properly. Numerous print options such as font inclusion, color control, page orientation, special printer features, and OPI can cause trouble at print time, yet the page will look great on screen.

These few comparisons reveal very different natures and uses of these two technologies and both have their place in a prepress workflow.



## **CURRENT ISSUE**

August 2002





# PennWell's Advanced Technology Division

publishes

more than 20

high-tech

titles & online

communities

The thing to be wary of is the multiple conversion steps of image files through your workflow. Consider the creation of the original PostScript file and conversion via the Distiller into PDF. Late in the workflow, the PDF file must be converted back into a PostScript file in order to be imaged. (Even a PDF RIP will need to execute PostScript code because of various OPI and PostScript code pass-through features.)

These conversion steps are extremely complex, especially conversion from a PostScript file into PDF using the Distiller. Things can get lost in the translation. (Remember all those Distiller options?)

The use of PDF for soft proofing is compelling and is probably the major use of the technology today. Remember that the PDF file is representing the original PostScript file that contains all the features necessary for high-end imaging. So, let the PDF file handle the soft proofing and the original PostScript file be used for the actual imaging. Trying to push the PDF file through another conversion can introduce additional risk.

The fact that some files will run through your RIP only when converted into PDF and then back into a PostScript file doesn't justify or mandate the conversion to an all-PDF workflow. PostScript errors can be diagnosed and fixed, but trying to convert everything to PDF will add unnecessary steps, errors, and complexity to most workflows.

#### PostScript preflight technology

As mentioned previously, PDF files can be used for soft proofing, but the Distiller can be further used for preflight purposes.

Most preflight technologies focus on the application file, prior to the generation of a PostScript file. While this is an important and vital aspect of the preflight process, the PostScript file is usually overlooked. Yet the PostScript file contains things that cannot be seen by the application preflight process, such as driver code, font inclusion, and OPI links. If the PostScript file doesn't image correctly, it doesn't matter what the application file looks like on the screen.

Thus, a preflight strategy which also looks at the PostScript file will provide the greatest printability assurance. In 1991, the first preflight product, known as LaserCheck, was introduced. This product allowed imagesetter files to be analyzed and rendered on a laser printer before the files were committed to expensive imagesetters. The concept is still valid today.

With the advent of desktop supercomputers another advancement of PostScript file preflight is possible. For the past 18 months work has been done with a major publishing company in which PostScript files can be thoroughly analyzed and a report generated. While some of the data overlaps the application-oriented preflight programs, analysis of the PostScript file, which truly represents the page image at its closest point to the RIP, is extremely valuable. It contains vital data that can be obtained only from the PostScript file itself, such as driver information, detailed image analysis, EPS module data, device-specific (PPD) information, etc. (Note that much of this valuable document structuring information is lost when a PostScript file is converted into PDF.)

#### PostScript file manipulation

As long as we are thinking about what can be done with PostScript files, consider modifying them to accelerate some of your workflow processes. Rather than considering the PostScript file as a necessary evil and a throw-away file, look at it as an opportunity to increase efficiencies.

For example, one system automatically inserts OPI links into yearbook PostScript pages, and completely avoids the need to place images at the application level. The insertion of special commands into PostScript files targeted at the Distiller is another great way to use PostScript files to control the Distiller parameters and add notes, URLs, or other features to PDF files. RIP-based OPI and font handling can be used to greatly simplify the submission of files to the production or proofing RIP. It is also possible to insert, extract, or replace sections of a PostScript file, such as bar code or price data, without going back to the original application.

#### Long live PostScript

I've heard some people proudly proclaim that PostScript technology is dead. Pardon me, but we've spent more than 15 years learning how to use this technology in our prepress workflows, it is the de facto standard for imaging today, and it can image the most complex pages. In my opinion we haven't even fully leveraged the power of Level 2 PostScript interpreters, much less all the cool things that can be done with PostScript 3.



Use PostScript and PDF technologies where appropriate. Both have a place in your workflow, but don't buy into the notion that an all-PDF workflow will lead to prepress heaven.

A well-informed staff, training, tools, and perhaps a bit of engineering will help you learn how to love your PostScript files and the extra profits that will surely come your way.

Frank Merritt Braswell, image linguist, is the president of Systems of Merritt Inc. (www.systemsofmerritt.com). He has been dealing with the PostScript language since the mid-1980s. His book, Inside PostScript, was published in 1989, and he has served the printing and prepress industry with software, training, and consulting services for more than 12 years.

## PDF vs. PostScript files

While portable document format (PDF) and the PostScript language are related, there are some fundamental differences and some nice similarities.

- Both share the same imaging model, which is great for converting from one to the other.
- PDF is not a programming language, it is a highly structured file format that carries page descriptions.
- The PostScript language has always had the ability to address the needs of high-end prepress.
- PDF was originally designed for screen-oriented business document portability. Prepress capability has been added since and is still evolving.
- PostScript files must be processed linearly, from start to finish.
- PDF pages can be viewed randomly.
- PostScript files can be executed (RIPed) on different platforms.
- PDF pages can be viewed on different platforms.
- PDF files are not always smaller in size than their original PostScript counterparts. In fact, sometimes they can be much larger.
- PostScript files contain document structuring conventions which provide valuable information about the structure and content of the print file.
- PDF files contain no document structuring conventions. Thus, for example, you
  are not able to tell where one internal EPS ends and another begins.—FMB

Electronic Publishing October, 2001



Search | Contact Us | Site Map | Privacy Policy | Bookmark This Site | Home

Copyright © 2002 - PennWell Corporation. All rights reserved.