TECHNOTE:
On the Importance of Print Testing:
A Brief Checklist

By Dave Polaschek

polaschek@applelink.apple.com
Apple Developer Technical Support (DTS)

Although the Macintosh printing architecture has only two major flavors — QuickDraw GX and traditional, Print Manager printing — many applications just can't seem to get printing right. (No fingers pointed here: you know who you are). For example, some apps require a specific driver in order to print correctly; other apps assume that certain undocumented APIs and structures will remain constant across different versions of printer drivers. (There's a reason we don't document LaserPrep, or how a driver uses the fields within PrintX).

This Technote provides you with a few, important guidelines to alert you to problem printing areas, and to help you test your application. If your application is guilty of breaking the rules, you should at least think about fixing the incompatibilities, since there will only be more change in the future.

Testing on a Non-PostScript Printer: Check #1

The LaserWriter driver supports sending PostScript to a printer within PicComments. If your application generates its own PostScript, you should verify that you can at least print to non-PostScript printers. Currently, the StyleWriter 2400 is a popular QuickDraw printer, so that's a good non-PostScript printer to test with, but the important point is to try printing with more than one print driver.

Things to look for:

■ do your images still print?

■ does text print?

■ does text print with the same line-breaks?

■ do you use entries from <<shudder>> LaserPrep?

In general, anytime you supply PostScript to draw objects on the page on a PS printer, you should also supply QuickDraw, so the same page can be printed on a QuickDraw printer. These things may seem obvious, but they'll make a big difference in how your customers perceive the usefulness of your app. While the customer may have a PostScript printer, it may be busy printing another job at the moment, or they may just consider it more convenient to print to the StyleWriter that's sitting on their desk. If your application will only print with a certain driver, make sure you're prepared to deal with customers who may want to print from another print driver.

One-pass vs Two-pass Printing: Check #2

LaserWriter 7.x and earlier implements one-pass (or direct) printing. It also supports two-pass (background) printing. LaserWriter 8 always does two-pass printing, as does QuickDraw GX. The two-passes through the printed data allow the driver to optimize the code that it generates for the printer. This comes at the penalty of needing enough space to store all the data on the system (boot) hard drive.

LaserWriter 8.4 will again support one-pass printing when it is released, so you should test with LaserWriter 7 with background printing off, and also with LaserWriter 8 to ensure the best compatibility with both printing methods.

There are two common mistakes applications make here. One is not handling errors correctly when a spooling print driver runs out of disk space (don't just put up an alert saying "Printing Error Code 575789390," use a descriptive message), and the other is depending on the driver somehow mystically divining what you really mean and doing the right thing as it optimizes your print job. This last falls into the category of depending on the specific behavior of a driver, but it's often unintentional.

An example is an application that "knows" that the LaserWriter driver will make a font available in PostScript if you print any character in that font, so the application will print a space character, and then ask for the font via PostScript. One problem is that the driver may not send the complete font when all you ask for is a space character. Another is that the driver may decide to name the font it sends something other than what you expect.

Only Make Print Manager Calls Inside PrOpen-PrClose: Check #3

An application should only make Print Manager calls between `PrOpen` and `PrClose`.

A lot of apps make the mistake of calling `PrintDefault`() with a closed driver. Although recent Apple QuickDraw drivers and the LaserWriter 8.4 driver are "smart" about this, older drivers still in use (such as LaserWriter 8.2.3f, LaserWriter 8.3.x, and Color StyleWriter Pro) are not. *Inside Macintosh: Imaging with QuickDraw* is vague on this point. In Chapter 9, the general sections maintain that the app must call `PrOpen`() before making any Print Manager calls, but the reference for `PrintDefault`() doesn't say explicitly that apps must call `PrOpen`() first. The documentation for all other functions states explicitly that `PrOpen`() is a necessary call.

The solution is to wrap calls to `PrDefault` with `PrOpen-PrClose`.

Make Sure You Check Error Codes: Check #4

Some apps aren't prepared for `PrStlDialog()`, `PrJobDialog()`, `PrJobDlgMain()`, or `PrValidate()` to return errors. With LaserWriter 8, if there is no chosen printer, the driver doesn't know what settings to use and returns an error. Some apps just keep right on going and crash because they have a nil handle for a dialog resource.

The solution is to check your errors and handle them gracefully.

Printing Based on QuickDraw GX: Check #5

QuickDraw GX has its own printing architecture, which is vastly different from the old way of doing things. More than likely, future versions of the Mac OS may be based on QuickDraw GX printing. At an absolute minimum, you should make sure your application can still print correctly with QuickDraw GX installed. A good minimum step to take is to use the QuickDraw GX printing dialogs, even if you still image via QuickDraw. If your application supports QuickDraw GX printing, you should test that more thoroughly, of course.

Summary

This has been a pretty short list of things to test in your application. Checking all of them in advance will at least let you know of and hopefully fix any compatibility problems your application faces now and in the future. It will also help you gauge the effort you'll need to make to get ready for Copland.

Note: This Technote will be revised and updated as new drivers come out and we move closer to Copland.

Further Reference

- Print Hints: "Top 10 Printing Crimes," *develop*, Issue 10.

- *Inside Macintosh: Imaging,* Chapter 9.

- Technote PR 10, "A Print Loop That Cares."

- "Adding QuickDraw GX Printing to QuickDraw Applications," by Dave Hersey, *develop*, Issue 19.

Acknowledgments