Creating PDF files for Printing and Archiving

Synopsis

This guide is designed to help you create documents that are going to be able to be printed accurately, with the printed output matching what you expected. As a side effect of generating files that print well, you will be generating archival or record documents that can be used effectively years from now. Your business is in creating the document, using the skills that you have mastered to provide information to your clients that is timely and accurate. This document is intended to primarily deal with architectural and engineering documents, although much of it can be helpful for any kind of PDF document.

The Issue

At Dataflow, we often see PDF files that are provided to us for printing that have multiple issues when it comes to printing a PDF file accurately. Some issues are simple to resolve, but others take a lot more effort, and some issues cannot be resolved without the assistance of the originator of the file. We understand the need to generate printed output that matches the output that you (as the originator of the file) intended. Errors in the printing process can cause costly errors, or cause potential safety related errors to go unnoticed in the construction of a structure. The PDF format is an extremely flexible file format, and as an interchange format, it has the potential to be ideal for exchanging information between parties, and as an archival record of what has been constructed. A PDF representation of the CAD drawing is a far better solution for long term storage and later reference than the original CAD file, since programs used to generate files change on a regular basis. As a result, there is no guarantee that 10 years from now, the original CAD files will be usable.

However, it is important to realize that there are several sub-formats of the PDF format, with certain types of files better suited for different purposes.

The Solution

The recommendation for creating PDF files for printing is to make sure that the files are in a true "print ready" format. In practice, this means that the files should have the following characteristics:

- Have all fonts embedded.
- Not be secured or encrypted.
- Be a PDF format that is also an ISO format.

- Be free of non-printable attachments.
- Not require any associated files.

The aspects that make a PDF file easy to print are the same aspects that make sure that the file will also be usable at any time in the future. By making sure that your files are set up to a standard, you will make sure that the files will print well, as well as make sure that when you need to reference those files 10 years from now, you will still be able to retrieve all the information that the file is intended to convey.

The easiest way to make sure that the files that you are providing to Dataflow are going to be printed accurately is to make sure that when you create the file, you specify that the file is generated in the PDF/A format. The PDF/A format automatically defines certain aspects of the file in such a way as to make sure that the file remains accessible in the future, and that all aspects of the file are printable. The "A" in the specification defines the file as an "archival" file, and that the file is specialized for the digital preservation of electronic documents.

To indicate the long term archival capabilities of the file format, PDF/A is also an ISO standard file format. As such, it is far more likely that years down the line there will continue to be applications that will be able to view and print the file. For instance, by forcing all fonts to be embedded, you avoid the issue of having used a font for the file that is no longer distributed with computers. A font that is not available in the file or on the computer will not be able to be displayed or printed, and over the years different versions of Microsoft Windows, for instance, have had different fonts added and deleted. This is a trend that we can expect to continue.

The PDF/A format prohibits many aspects, including audio and video content, JavaScript and executable extensions, encryption, external content and depending on the version of the format, certain compression formats as well as transparency. The format does ensure that fonts are embedded, that the colorspace is well defined, and that the metadata for the file follows standards.

Due to the prohibition of JavaScript and executable extensions, the file format is also safer, in that it is more difficult to have the file be a conduit for malware.

The same aspects that make the file format a good candidate for long term storage also makes the file ideal for printing. Non-embedded fonts, for instance, require that the font be loaded from the system that is used to print the file. In certain circumstances, even if the "same" font is found on the computer that is used for printing, the font may not be an exact match for what was used in creating the file. In other situations, when the font is not available on the computer that is used for printing, a substitution has to be made to allow the file to print. In some cases, the substitution will not show any ill effects, but in other cases it is possible that entire letters, words or even entire contents of the file may be missing. At the very least, a substitution will generally cause formatting changes in the final output, which may make other text unreadable due to overprinting, for instance.

Exceptions

It is important to note that even though you are using a PDF/A format, or you have selected the option to embed all fonts, it is still possible that a certain font will not be embedded in the file. This is because

of licensing options for certain font types, which will prohibit sharing or embedding those fonts. Most applications should warn you about such a situation, but it is always worth checking to see that the font was in fact embedded into a file. Using a PDF reader such as Adobe Acrobat or Acrobat Reader, you can go into the Properties section of the file (typically under File \rightarrow Properties or accessible via the Ctrl-D shortcut) and view the fonts section. If you see that some font has not been embedded, you may want to select a different font for your text in the file, since this font face **will** be substituted when printing from a computer that does not have that font installed. If you see a problem such as this when generating a file from a CAD application, you may also select to have the text rendered as shapes or geometry when generating the PDF. This way all the text is converted to actual graphics, and will print the same way as any other line type would from the original, although the editability of the text may be lost.

Summary

The PDF/A format is ideal for both archiving and printing, and can usually be generated by any application that is capable of generating a PDF file. Most applications that are capable of generating PDF files as output will have an Options section, and will then allow for the choice of PDF sub-formats to be used. If this is available, you should select the PDF/A format when you go to generate the file. If that is not an option in your application, please make sure that you have indicated in the options that you want to embed **all** fonts into the document as a bare minimum, and that you have not selected encryption of the file.

In short, when possible, generate PDF/A output from your files when printing or archiving is desired from those files. If is not possible to use PDF/A, make sure that ALL fonts are embedded, that you are not using transparency, that the file is truly standalone, and that you are not using encryption or security settings on the file.

Further Information

Should you have any other questions regarding this document, please do not hesitate to contact your local Dataflow office. You can find contact information at our website, located at <u>www.goDataflow.com</u>.

Wikipedia (<u>www.wikipedia.org</u>) has some great information and further pointers to PDF file formats at <u>www.wikipedia.org/wiki/Portable_Document_Format</u> (or search for PDF).

Dataflow print anything. document everything