

Adobe Acrobat and its Significance to the Publishing/Print Production world.

Adobe systems' offering to the publishing world is heavily based upon the foundation that they created with the infusion of PostScript into the professional community. This quiet revolution that has taken place over the past decade has given Adobe Systems the unique "Cat-bird Seat" about this whole PDF (*Portable Document Format*) because they already have an extremely large (industry standard) installed base of PostScript assembly language users. Meaning that the engine that is Acrobat (PostScript) is already what the users are using to create their files *Today!* They need change nothing in their current workflow or learning curve in order to produce accurate Acrobat PDF files. They use the same programs... proof the same way, and compose their documents *exactly* the same for both PDF documents and for print. That is how Adobe is making the transition completely painless to current PostScript users. This is the foundation that has allowed Adobe Systems to put their foot in the door so to speak in relationship to allowing users a painless way to compose both desired results (print and PDF) in the interim timeframe while PDF gains acceptance and market share. This PDF technology easily lends itself to CD ROM distribution (for the near-term future) and to the Internet and other "Information Superhighways" that are slated to the masses. The reason I am downplaying the significance of CD-ROM is that it is really only a "Quick-Fix" for the problem of moving, and distributing large amounts of information to a desired user-base in a cost effective manner. CDs are slow, non-writable, non-erasable and short lived for the long term because they are not easily written and not erasable. CD-ROMs are great right now because our current Telecommunications/Internet situation... "The Information Dirt-Road" does not have the bandwidth or capacity/speed in order to provide efficient access to all users. It is much more efficient to mail a ¢.99 CD-ROM (650 Megabytes of data) to an end user who has a low-cost CD-ROM drive than it is to try to attempt to modem or telecommunicate the same amount of data to the user who probably does not have the drive capacity to hold all 650 MB's if they could get it modemed in anyway. CD-ROM, while being a very useful tool for the present... will become obsolete by the power of what will become the rails for the information "Superhighway" in the next few years... Fiber Optics. Fiber will allow the kind of transmission rates that make this type of data distribution to the masses possible. It is estimated that in three years... Fiber will become available to 80% of America's households. Within this timeframe, personal computer power will also have grown to the point that manageability of large amounts of files will also become a more attainable ability of the basic computer user.

Enter Acrobat... The great equalizer...

Acrobat makes the process of "Portable" documents possible by "Distilling" the components that make up any PostScript page into a compressed version of the page. Any PostScript typefaces used, any scanned images and any placed EPS graphics are distilled, embedded, and compressed into the final "PDF" document that is the page. This PDF page rarely exceeds 200 Kilobytes (a file that can be modemed in a couple of minutes under current modem technology... and about 1/10 second with fiber) and can be printed to virtually any printer. (More on that issue later.) This means that the recipient of the document need not have a 40MB Quadra/ Power PC (and an \$8,000.00 typeface library) to properly view, read, print and enjoy the page involved. The added advantage to the PDF document is that the entire text is searchable and indexable. This feature (searchability) gives the PDF file some distinct advantages over paper. A user can search an entire folder of documents on any criteria, and obtain a listing of PDF documents in that folder that contain the search criteria... organized by a "relevancy" rating... ie: how is the information contained in the PDF file to what I am searching for? This extremely powerful aspect to the system is what makes it in some respects "better than paper." Now, before everyone jumps up and says... "Wait just one minute you young Whipper-

Snapper... Printed pages will never disappear!”... I agree. What will change is the production/distribution methodology that we have gotten so accustomed to. Traditionally, pages are composed by assembling the elements that make up the page and sending them to the separator/engraver for film separations. These film separations are then sent to the printer who burns plates, makes impressions from the plates, trims, saddle stitches and distributes the final bound publications to a distributor, who then mails the final publications to the paid subscribers via the U.S. Mail system. This workflow not only incorporates millions of dollars of labor and materials... but incorporates a large opportunity for human error and mistakes... the more processes, the more chance of someone making a human error. Errors mean unexpected delays and extra costs. The Acrobat workflow stems around generating documents that are ready to go to the end reader immediately following approval at the site of creation. This means total and complete control over distribution can be maintained by the publisher should they desire. The publisher no longer must deal with third-party vendors of supplies and services in the future. By merely generating the PDF documents, the page is still not printed. the printing of this page will occur not prior to distribution as the current model suggests... but at the end user’s site... and on their personal printer. That is where the next piece of technology enters that game... marking engines.

Enter the printer... QuickDraw comes of age.

In the mid to late Eighties, Apple Computer introduced a very futuristic software module called “32 bit color QuickDraw.” This software was a system extension that allowed any program running under the operating system to display as many as 16.7 million colors... at the resolution of the device. By making this system extension part of the system software (and eventually part of the ROMs or Brains of the computer) Apple ensured that all software would have the capability to do so. This meant that what you could see on your monitor was now closer than ever to being WYSIWYG (What You See Is What You Get) if you were printing to a non-PostScript printer like an ImageWriter II. Any time your monitor showed you a page, a character, a program or anything... it was displayed in QuickDraw on the monitor. Printers like the ImageWriter use QuickDraw to generate the bitmap that they can print. These “Dot-Matrix” printers like the ImageWriter were “Dumb” printers because they lacked a brain that figured out the optimal image for the print job and generate a bitmap using it. These printers were slaves to the computers... and prior to QuickDraw, the output from these printers was spotty at best. They could only reproduce bitmaps of type characters and graphic elements. After the introduction of QuickDraw however, this started to change the look of graphic elements printed on these “Dumb” printers. All of a sudden... rules got more accurate, ovals got more rounded, and page elements got more precise in relationship to the page itself. QuickDraw was the brain that the printers lacked... and although had to run on the computer itself, it was capable of displaying at the maximum resolution of the device. This meant that what looked good on the screen at 72 DPI, also looked as good as the printer could print... at 144 DPI. etc. Ironically, this language called QuickDraw is very similar to PostScript in many respects. Both are named “Description Languages” because they basically “Describe” the page, screen view, or image to the respective device... (The printer for a page, the monitor for a screen). There is a clear difference between the jobs that both of these languages do. PostScript is a language designed to describe pages to high resolution printers... a procedure that requires calculations that are far more complex and precise than the requirements for a computer desktop. QuickDraw was designed to perform its calculations very fast (hence the name “QuickDraw”). This was imperative to the Apple engineers because everything that is displayed on the monitor of the computer must go through a QuickDraw routine... If the engineers incorporated too many features and abilities to QuickDraw, it would soon cause the computer to have “Not So QuickDraw” performance. For this reason, QuickDraw just

performs the rudimentary tasks of displaying what the program has to say... as fast as possible... and that's it.

Adobe Type Manager... the quiet little sleeping giant...

Up until this point, as mentioned, the two languages bouncing around peoples computers (QuickDraw and PostScript) lived in their own little worlds... PostScript in the Laser-Printer world, and QuickDraw in the monitor/dumb printer. This scenario seemed to work well enough except that since all of the cool typefaces were written in PostScript, they really did not look all that good on the screen. This was because there was no way that the computer (QuickDraw) could read the PostScript "Printer" font because all it knows is QuickDraw. This meant that there had to be two versions of each font... a Printer font... (a PostScript printer font) and a Screen font... (a screen "Bitmap" of what that font might look like on the screen at a certain size) that QuickDraw can display when called upon. The real disadvantages to this are that the "Screen" font becomes the only thing that QuickDraw can display and that follows also to QuickDraw printers... (The "dumb" printers that can only print the dots that the computer can generate.) This makes the output of a screen font from a QuickDraw printer much lower resolution than the printer is capable of imaging... because the screen fonts are meant to be a best case bitmap for the monitor... 72DPI, whereas even the lowest end QuickDraw printers (The ImageWriter) print at 144DPI. So the computer world had a vast difference between the types of printers that were able to be used for certain applications... (QuickDraw and PostScript) and frankly, the QuickDraw printers, while being affordable, left a lot to be desired in print quality... and not because the printing engine did not have enough resolution to render a sharp character, but because the QuickDraw versions of typefaces were designed at 72DPI... and did not care about printers. When printing to a PostScript printer (a printer with a computer that prints PostScript typefaces and graphics with a computer *in the printer...*) PostScript Printers have a fully self-contained computer within the marking engine called a "PostScript Interpreter" or RIP (Raster Image Processor) that generates the bitmap that the marking engine (usually a Laser Printer) can print on paper. This "PostScript Interpreter" is the program that is run by the RIP to make this high-resolution bitmap that the printer can print. So, in PostScript, the characters are not even generated until they hit the printer; whereas in QuickDraw, the computer generates the characters in the computer... and sends the bitmap to the printer.

Adobe Type Manager came out in about 1988 (From Adobe Systems). This software revolutionized how the software applications displayed typefaces on the screen by performing some of the duties of the PostScript Interpreter... in QuickDraw on the screen. Adobe Type Manager or "ATM" as it has affectionately become known, looks up the PostScript fonts from the hard disk... and translates or "RIP's" them into QuickDraw routines for display in the program. All of a sudden, WYSIWYG *really* meant "What You See Is What You Get". ATM was bridging the gap between QuickDraw and PostScript by serving as a PostScript Typeface translator for QuickDraw. Although this software gained wide acceptance for its ability to make the screen look as close as possible to the final page, its most striking and un-sung feature was that of its ability to render PostScript typefaces on Quickdraw devices other than the screen... printers. ATM allowed QuickDraw to render fairly high-resolution characters to low-cost QuickDraw printers... and these printers had been getting better over the years as technology moved away from the "Dot-Matrix" technology to the "Ink Jet." Ink jet technology promises to deliver higher and higher resolutions with a low unit cost per printer... and decreasing costs per printed page. Current Ink jet printers are capable of printing 360 to 720 DPI in 4-color process inks... resulting in extremely high quality photo-realistic images, as well as exceptionally crisp clear type... for less than ¢.15 per page. In addition, these printers are now available in color for as little as \$600.00 and as little as \$200.00 for black and white. It is my belief that the development

of this generation of marking engines... low cost, low cost per page, high quality, QuickDraw printers are going to be the vehicle that allow the whole PDF revolution to take place. These printers are capable of generating near SWOP quality pages... in the home or office... with consumer level devices and supplies/consumables. The trend in ink jet technology is only moving towards higher resolutions, better color, and more nimble speeds. The key to this movement is the full and complete bridge between the PostScript and the QuickDraw languages... and that bridge is Acrobat. There is really nothing special about these new printers... they are still “dumb” by depending upon the computer to generate bitmaps for them... except for their exceptional quality and their low cost.

What role does Acrobat play in all of this mumbo-jumbo?

The really special niche to this scenario is that Acrobat puts PostScript Interpreter software onto the computer itself... and not in the printer. When you generate a PDF file from a document... you first generate a PostScript file... that is then “Interpreted” in the Acrobat Distiller (an Adobe, Configurable PostScript Software Interpreter) into a PDF document. The PDF document contains precision placement information to all elements on the page as well as their PostScript typeface, raster (scan) and vector information... all compressed and embedded into the resultant PDF file. The PDF file that comes from all of this is usually extremely small in size and therefore extremely compressed. This is a configurable setting however and a “Lossless” compression scheme can be used instead. This ability (the one to NOT compress the image with a loss) is one striking capability of this program. It is theoretically possible to use the Acrobat software interpreter to include high-resolution 4-color image data and produce high-quality separations. The current Acrobat print dialog box does not support printing separations... but it is possible. Adobe has also built their “Plug-In” architecture into Acrobat 2.0, allowing upgradeability and modular functionality approach to the software. This open architecture will allow a third party developer (or Adobe Systems even) to develop a software “Plug-In” that would allow the printing of separations to imagesetters. The real beauty of this is that it could help to establish a standard among the entire pre-press industry by only requiring the final film/print provider to use one program... Acrobat Exchange... and one file, a PDF. Using a lossless compressions scheme to generate the PDF file will result in a large file of course, but no larger than the original high-res data, and the convenience of only one program and one file have some tremendous allure. The other advantage to this system is that due to the device/resolution independence of PostScript, the same final file will work for a variety of devices... without re-generation of the print file. A color proof of a document can be pulled prior to film... and the exact same file can be used for final separations. The promise of standardization is one of the most powerful ammunition brought out by Adobe Systems in the goal of cross-platform compatibility. While there is considerable effort being exerted by panels such as TIFF/IT, IT.8, etc. to establish one file format that will be suitable for all vendors of CEPS systems (Scitex, Crosfield, DS Screen, Linotype-Hell); most of these factions do not consider PostScript to be an acceptable standard between platforms for a variety of personal issues. (My own personal belief in this is that the proprietary vendors do not want to have to adopt a platform that they will have to license from Adobe Systems, hence their resistance and nay-saying.) Moving to PostScript is going to have to be the logical step for the simple reason of the explosion in popularity of PostScript composition systems on the desktop computers. The one thing that all desktop composition systems have in common is that they output PostScript files (either to a printer... or to a storage volume). It is because of this I feel as though PostScript will be the next logical step in standardization, and the “Portable” (PDF) version of its documents will help to establish that foothold.

Another of Acrobat’s abilities is to allow DOS, Windows, UNIX, and Macintosh systems all to share the same file. This “Platform Independence” feature begins to break down the walls that that have

long divided the desktop jungle.

And Finally... At no additional cost!

Using Acrobat's linking tools (allowing a user to click on a text field and lookup another PDF document or directory) the potential exists to use Acrobat as an Internet "Browser". A browser is software that help to automate navigation around the internet (more specifically, the World Wide Web... or "Web Browser") and its associated databases. Currently, one of the most popular "Web Browsers" is developed by the NCSA (National Center for Supercomputing Applications) at the University of Illinois, Champagne-Urbana. This program (available in Windows, UNIX, and Macintosh versions) called "Mosaic" is a very flexible "HTML" (Hyper-Text matching language) that is used to browse through the World Wide Web (WWW) sites with the ease of pointing and clicking on text fields and graphics. This revolutionary program started a movement to generate internet sites that read and displayed HTML pages for use with NCSA Mosaic. Gradually, more and more sites have added HTML/Mosaic compatibility, which greatly adds to the ease of use and navigation around the vast resources available on the WWW. Due to the platform independence of Acrobat, the ability for it to be used for an internet browser is a very real thing. Perhaps a plug-in could be developed to allow the links to link to other PDF documents available on other Web Servers. The beauty of this is that high-quality pages (pages that look great on both screen and printer) can be distributed to remote users via Web Servers, and also provide an interface for internet navigation. Although not a present ability of Acrobat, the potential is there, and should acceptance come about, (due to Adobe Systems offering the Acrobat Reader v. 2.0 *FREE* to anyone interested.) this system could be just the standard that the internet now desperately needs. The potential of this software in this role will not be fully realized until a general consensus and comfort level is reached by the Web Site Community; a group of folks notoriously afraid of companies with a financial interest in such a project (Adobe Systems) and could possibly resist just because of that reason alone. Time will tell, but this may be a feature to watch as distribution of high-quality publication pages via the internet becomes more and more the norm.

Now repeat all of that for me....

In short Acrobat is the first attempt at true standardization among computer users and platforms for the printed page. The potential exists to use the system for high quality separations, as well as proofs. The system can also be used for Remote Proofing. In the situation where print media production takes place off site (or even cross continent) at a design firm/separators can generate an Acrobat PDF file of a page, and rather than the fairly rough looking FAX proof in B/W only, the PDF file can be modemed to a small computer/color inkjet printer at the corporate site for output in a matter of minutes over normal phone lines. This workflow could be used as a color PostScript FAX... incorporating a two or three minute phone call (because of the small size of the PDF file) to send the file that contains all of the typefaces and graphics used. The receiving site need not have a ton of fonts or memory... just a low-cost color printer, and a modem. This might allow a decrease in FedEx charges if the resulting printout can be discussed in detail via telephone... or "Post It Notes" can be added to the PDF file and it can be re-modemed back to the other site. Another excellent use from Acrobat will come from the ability to distribute high-quality pages to remote users via the Internet or other on-line source... in a very timely manner. Some of the other uses of Acrobat that I have outlined here are realistic, and needed, but may require some time to implement. There is *no question* however that this program has just begun to impact the printing/publishing industry and should be watched very carefully.