



## SGML Versus Acrobat: Which to Use for CD-ROM/ Online Publishing?

**Among the most promising contenders for digital-document-delivery dominance, PDF is far and away the leader.**

**h**ere's a riddle: What do the electronic publishing efforts of the Internal Revenue Service, Intel Corporation, and the Vatican have in common? The answer: They all use Adobe Acrobat's Portable Document Format (PDF). And there are many good reasons why they do.

Among the most promising contenders for digital-document-delivery dominance, PDF is far and away the leader. PDF is certainly not alone in offering a medium which acts as an excellent legal archive, since the delivery medium is read-only and therefore cannot be edited and must always be produced from another source. But unlike the other contenders, Adobe Acrobat's PDF is a *de facto* open standard, based on PostScript. And no less a heavy-hitter than Dataware has developed CD-ROM production and navigation systems built to work with PDF.

### **THE PLEASURES OF PDF: EASY TO LEARN AND EASY TO USE**

PDF's new medium is easy for authors and users to like. With a simple print driver, from any application on any major platform, you can "print" to PDF. What's not to like?

The format delivers virtually perfect color renditions of paper originals, and virtually perfect renditions of fine layout details, fonts included. It also supports the easy creation of hypertext links. Furthermore, using Acrobat requires essentially no setup, nor any training to speak of to produce PDF or to use it. In essence, if you can print, you can produce PDF.

Not only is PDF easy to produce, it is binary-compatible on all major platforms. You can produce PDF

on WinTel platforms, Macintosh systems, and UNIX; and you can view and use the results on DOS and the production platforms. The viewer, called Reader, is freely distributable. And PDF documents can exceed E-size CAD drawings, up to 45 square inches.

Combine this amalgam of features with PDF's potential electronic paper dimensions, and many interesting opportunities arise.

For example, all your architectural CAD drawings could become a Geographic Information System. A typical search might say, "Find me all our properties located near North Central Avenue in St. Louis"; presuming "North Central Avenue" is a text annotation in the drawings, Acrobat would deliver on the request.

If that isn't enough, Adobe has made its version of digital paper multimedia-savvy; building a QuickTime video sequence is a simple drag-and-drop operation. With a plug-in from Verity, PDF offers powerful full-text searching combined with structured searches. In minutes, you can index vast collections of PDF files and freely distribute them to others. The search engine itself comes bundled in Exchange, which also contains the PDF-writer.

And PDF is suitable for CD-ROM/online hybrids, Web-enabled, right out of the box. On WinTel platforms, compound document creation is a snap with OLE objects. You can easily create CD-ROM/online hybrids of virtual documents spanning the globe. Even the freely distributable Reader can link to Web hypertext nodes within a WWW site, each node's content changing at will.

#### **PITFALLS OF PDF: PAPER-LIKE LAYOUT LIABILITIES**

PDF's virtue can also be its Achilles' heel. PDF duplicates paper so well that the end result can suffer from paper's limitations. Paper layouts often do not work well in electronic media. However, it is true that your composition system, with the flick of a style library, can add navigational cues at no cost. For example, you can apply a style library that uses various screen colors for different types of section headings. This can add visual cueing to on-screen PDF renditions. Still, the result is an essentially flat file.

Whatever hypertext structure you add to a PDF file must generally be done by hand. Although there are third-party products to generate links from binary codes embedded within word processor files, these are guaranteed to be problematic since word processors change their binary conventions at will and users are not constrained to use them correctly. Make one small change to a word processor document, print to PDF, and you must recreate all the links anew.

Acrobat's flat nature becomes evident when text searching multicolumn documents. A search for three words in proximity may yield words in three adjacent columns, or words in the bottom of a column or in a page footer—two totally unrelated segments.

Easy and very powerful, yes. But will it hold up to very large production needs where structure is important? Will PDF still suffice when the sheer

size of your document base makes the importance of structure rival the significance of its content?

#### **TAKING THE LONG VIEW WITH SGML**

Some riddles have two solutions: In addition to PDF, the IRS, Intel, and the Vatican *also* all use SGML. SGML is a Federal Information Processing Standard, and appropriately enough, the IRS was one of its early adopters. Intel, along with a consortium of other integrated circuit vendors, paid a vendor to create the PCIS Document Type Definition (DTD) to allow their customers to view and compare information from chip vendor catalogs; and the Vatican library has a project underway that has encoded all the Reginensis manuscripts using the Text Encoding Initiative (TEI) DTD architecture.

SGML's payoff is an annuity stream that lasts for a long time. However, no one says the initial investment in SGML is cheap or easy. You must spend energy to analyze your present and future document needs: What kinds of documents are you likely to want to derive from your document investment, and what navigation aids will you want to automate?

Moreover, SGML is the gold standard archival format for your content. You will have insulated your document base from the vagaries of tools, media, and vendors. Do you want to bet your document base—or, as with scholarly collections, the patrimony of our civilization—on any given vendor's next release or whatever will follow HTML version 3.2?

One legal publisher reports a recent experience with PDF that gives a sense of the impact that making small changes can have when a heavy investment in hypertext links is made in a non-SGML, proprietary format. In this case, the changes—while quite minor—required all the links to be recreated. With a smaller document this would not be a problem, but with a large or frequently changing document collection, it can make rework costs unacceptable. SGML-based tools make hypertext maintenance just another feature.

Remember the key advantage of PDF, that it "looks just like the original"? If that original doesn't work well on a 75dpi screen, you may not realistically be able to produce large volumes of electronic documents in PDF without significant

reformatting of the original source documents. You may not be convinced of the need to preserve your option of unlimited opportunities to express the original content, but can you be sure you now know all the uses to which you may want to put that content investment?

It also is increasingly clear that searches of large volumes of documents are requiring ever more refined ways of filtering the searches. It's getting more and more difficult to pose a good search query at a major Web site such as Alta Vista without getting thousands of hits. SGML can focus your customers' searches and eliminate the false proximity hits caused by adjacent columns and footers.

On the other hand, SGML isn't as easy to manage and manipulate as PDF. It does require a disciplined commitment to analyze your needs and develop DTDs accordingly. Preserving the exact look and feel of page layout may not be possible in some cases with SGML, and certainly it is never as easy as with electronic paper media such as PDF. An emerging set of SGML systems—spurred by the growth of HTML and the Internet—will raise the bar for those who ignore the benefits of this investment.

#### **ENVISIONED SGML/PDF HYBRIDS: THE BEST OF BOTH WORLDS?**

For several years now, we have heard that Adobe is planning to make Acrobat SGML-aware. In fact, Adobe's recent OCR product, Capture, has rudimentary block capabilities. For example, it recognizes text arranged in bulleted lines as lists.

Could this presage a structured layer within Acrobat which could combine the structure of SGML with the ease and presentation of PDF? If Adobe succeeds in mapping form back to content reliably, this could mitigate the rework of files by preserving automatic hypertext links. Perhaps, with the potential convergence of PDF and SGML, we could soon find ourselves witnessing a synergy of benefits in yet another new age of electronic publishing systems.

-----  
*Robert J. Boeri and Martin Hensel are columnists for INFORMATION INSIDER. Boeri is Advanced Systems Specialist in the Information Services Division of Factory Mutual Engineering of Norwood, Massachusetts. Hensel is founder of Martin Hensel Corporation, a Newton, Massachusetts-based consulting firm that builds SGML-based editorial and production systems for publishers, corporations, interactive services, and compositors. ■*