

Journal of

INDUSTRIAL TECHNOLOGY

Volume 21, Number 2 - April 2005 through June 2005

Challenges Ahead for Adobe's Portable Document Format

By Dr. Paul Cesarini and Mr. Jason Mellen

Peer-Refereed Article

KEYWORD SEARCH

*Computer Technology
Graphic Communications
Information Technology
Internet
Philosophy
Visual Communications*

The Official Electronic Publication of the National Association of Industrial Technology • www.nait.org

© 2005



Dr. Paul Cesarini is an Assistant Professor in the Department of Visual Communication & Technology Education (VC&TE) at Bowling Green State University. His current research and work focuses on digital rights management and the on-going erosion of fair use for audio, video, and text. Prior to teaching in VC&TE, Paul created and managed the BGSU Student Technology Center, Laptop Loan Program, and annual Technology Fair. He reads Wired far too much.



Jason Mellen is a student of Visual Communication Technology and a Digital Media Technologist with Bowling Green State University's Digital Media Research Group. Jason's research and work focuses on the areas of digital media, e-Learning, and computer-aided collaboration. He also serves as the resident advisor to IMPACT, a living-learning community focused on the integration of moral principles and critical thinking.

Challenges Ahead for Adobe's Portable Document Format

By Dr. Paul Cesarini and Mr. Jason Mellen

Introduction: the Promise of PDF

Universal document exchange, that elusive goal only slightly less far-fetched than the paperless office, has largely become a reality over the past few years. Adobe's Acrobat line of products, recently released in their 7.0 incarnation, have steadily improved and have achieved a near-ubiquity in a variety of industries and disciplines. The publishing industry continues to rely on Acrobat for all things print. Web content developers and technical communicators rely on it for convenient and efficient ways to make manuals and related documentation accessible to end-users. Adobe has recently begun pushing Acrobat as an essential tool for enterprise, as well, by promoting it as a foundational tool for accessing and sharing corporate data.

Creating a portable document format (PDF) file in many cases has now become as easy as printing, and often involves little more than a single mouse-click. This is a far cry from earlier versions of Acrobat that often involved complicated distillation processes and intermediary file formats to get from the original to a PDF. Adobe has also made it increasingly easy to transfer a PDF file to a Palm OS or PocketPC-based device, which further enhances the portability and appeal of the format.

This increased ease of PDF creation, and corresponding overall popularity of the PDF format itself, has not gone unnoticed. In *Portable Document Format (PDF) – Finally, a Universal Document Exchange Technology*, author Wan-Lee Cheng espoused the likelihood of long-term sustainability and growth in popularity of PDF. He states that, "Although PDF is still growing, there is no doubt

that it will become the mainstream technology for electronic communications, document distribution, and printing / publishing workflow." (p. 61) While the author agrees with his overall premise that the PDF is robust, mature technology with great potential, it may not be quite so clear-cut.

Despite overall gains in ease-of-use and a broad user base, the PDF now faces a series of hurdles and roadblocks toward its very universality. These include the potential for malicious code to be introduced into PDFs, increased usability concerns associated with PDF-based content online, the continued lack of standardization in the electronic book (eBook) industry, overall readability on computer screens or web browsers, and stiffened competition from several key players in different industries, including Microsoft, Macromedia, Palm, and Autodesk.

An Open Specification, a Broad Base

How has the PDF managed to become so widely used in such a relatively short period of time? The PDF is barely over a decade old, yet its reach is both far and wide. The open licensing stance taken by Adobe has a great deal to do with the success of the format. Adobe does not charge a fee for their PDF format. Adobe has often stated that "the PDF is available to anyone who wants to develop tools to create, view, or manipulate PDF documents," and has gone so far as to openly publish the PDF specification, in order to encourage development of third-party PDF solutions. The following legal notice from Adobe's Developer Solutions site relates to this openness:

Adobe desires to promote the use of PDF for information interchange among diverse products and applications. Accordingly, [seven] . . . patents are licensed on a royalty-free, non-exclusive basis for the term of each patent and for the sole purpose of developing software that produces, consumes, and interprets PDF files that are compliant with the Specification (Adobe, 2003)

These actions have allowed the PDF to flourish in Open Source Software (OSS) products such as the free OpenOffice.org suite, which allows for on-the-fly PDF and Flash conversion with any documents generated by its word processing, spreadsheet, presentations, or web-authoring components.

OpenOffice.org Community Manager Louis Suarez-Potts clarified this license somewhat. According to Suarez-Potts, while not necessarily an open standard, which often implies a peer review process, and certainly not open source, PDF is indeed an open specification. (Louis Suarez-Potts, personal communication, January 14, 2004) Adobe spokesperson John Cristofano further added to this idea in Mozilla Magazine. He stated that "PDF is a published specification - meaning it is publicly available for anyone to acquire; anyone who acquires the specification has the capability to create programs that read and/or write PDF files." (Angelo, 2003, p.2)

This openness and broadly-installed existing user base were likely deciding factors behind Steve Jobs' decision to base the 2D rendering library of Quartz, the primary imaging model of Mac OS X, on Adobe's PDF format. As a result, PDF is the default file format of Mac OS X, and anti-aliased text and graphics within Mac OS X is significantly improved from previous versions of the Mac OS. Additionally, OS X users can generate a PDF output from within any application, rather than hardcopy, without having any version of Acrobat installed.

While this degree of native PDF conversion is currently not possible

on Windows-based platforms, and in all likelihood Microsoft will never opt to incorporate this feature, Adobe has compensated for this by releasing a barebones version of Acrobat exclusive to Windows, called Acrobat Elements. For about \$50, Acrobat Elements provides a minimalist feature set that effectively gives Windows PDF feature parity with Mac OS X. Adobe touts this basic level of PDF functionality as being a way for businesses to "extend the value of their Microsoft Office investment by standardizing on Adobe Portable Document Format (PDF) for reliable document distribution." (Adobe, 2004) Third-party companies have also made in-roads here, as well. Companies such as Xelerate, Data Becker, Scansoft, and others have released inexpensive software for Windows that allow basic PDF creation. The open source community has also stepped-up in this area, by developing free PDF generators like PDFCreator.

The Strength of XML

Recently, Adobe has taken steps to achieve a new level of openness with the integration of the Extensible Markup Language (XML) data format into their PDF platform. A new meta-format called XDP (XML Data Package) combines traditional PDF files with XML markup that can contain both information from the document and information about the document. (Adobe, 2003b) XML is a multi-purpose file format for the representation of "data objects." To understand XML, it is perhaps helpful to first think of Hypertext Markup Language (HTML); the language of the web. HTML "tags" are used to indicate various styles and attributes within a document. However, HTML contains a limited vocabulary that is understood by web browsers for displaying documents. XML, in contrast, has no specified vocabulary. Rather, it is a file format to which various, context-specific, vocabularies may be applied. (DuCharme, 1999) For instance, the insurance industry has a standardized vocabulary, ACORD, for exchanging information in XML files while healthcare organizations utilize the HL7 vocabulary. (Adobe, 2003b)

XML has certain advantages over other binary data formats. For one, XML files are human readable in that the content is understandable in its raw format without special machine formatting. Additionally, because every XML document conforms to a given structure, the data can be exchanged and validated using generic XML parsers and exchange systems (such as XML-RPC). (DuCharme, 1999) The open nature of XML data makes it easy to share single data sets among different applications. An example would be a meeting agenda for an enterprise corporation that contains XML information about both attendees and the meeting itself. A calendaring program would interpret the relevant XML in order to add the meeting to a user's calendar. Meanwhile, a user's contact manager would parse information about the attendees and add that information to an address book.

Given the powerful nature of XML, Adobe's adoption of this data format will likely push the PDF format in new directions and take the platform into uncharted waters. Adobe's XDP files will use XML to represent four types of information: form templates, form data, configuration information, and other information such as digital signatures and meta-data as well as the traditional PDF data. Form templates provide the technical descriptions and field-mappings used when linking XML data to form fields in the PDF documents. The form data is the actual vocabularies (like ACORD) that are used in the document's embedded form fields. The XML configuration data is responsible for various databases and SOAP connections for that document. SOAP is a protocol used for exchanging information with web service applications.

The ability to have information contained within PDF documents exchanged with web applications is what has the most potential to reposition the format in the marketplace. Imagine a product distributor issuing a Request for Proposal in an XDP document. Each supplier would have a customized document based on the XML embed-

ded in the file. That supplier would then fill out the RFP form and select the submit button. Processing instructions within the document would validate the information and then send it to the computer systems of the distributors. The final products would still be exact copies of the RFP forms complete with the updated data. However, the data has already been sent and processed by the distributor by the time the supplier could even hit print. (Adobe 2003b) The PDF framework also has security measures in place to allow for encrypted exchange of information as well as the authentication of users via digital signatures. Adobe has developed partnerships with leading digital security firms such as RSA, VeriSign, Entrust, and ActivGaurd. (Adobe, 2004b) The addition of XML data into PDF will move the format from that of a mere transporter of static documents to a platform for the exchange of enterprise information.

The Trouble with Universality

This ongoing integration of XML into the PDF specification will most likely only help make the PDF itself become even more widely diffused, with an even broader user base. Yet, while this broad user base may be the strongest asset of the PDF, it also has the potential to be its chief source of problems. Rob Lancaster, a technology research analyst for The Yankee Group, recently voiced concerns about the PDF. He stated that “a big part of that appeal [of PDF] rests on the ubiquity of the viewing capability.” (Becker 2003) That is, the ubiquity or universality of the PDF is its primary strength and its primary appeal: it is universal, so it is widely-used; since it is widely-used, it is universal. To appropriate an agricultural term, PDF appears to be steadily evolving into a monoculture

What, then, is the problem? If a critical mass of users adopt the software or file format of a certain company, which in turn forms a defacto standardization within different disciplines and/or industries, would not such standardization ultimately be for the best? Pos-

sibly, but not necessarily. Technologist Richard Hoffman perhaps best explains why such a technological monoculture should be avoided. In *Resisting the Path of Least Resistance*, Hoffman states:

A monoculture tends to be fragile because of its sameness. Perhaps more importantly, however, a monoculture is by nature slow to adapt, change and innovate--particularly in the case of the computer industry. What incentive is there to really innovate when your only competitor is yourself? All you need to do is make slow, incremental changes just often enough to assure a steady supply of upgrade revenue. (Hoffman, 2001)

Of course, continued innovation in the PDF monoculture has not yet stalled. In addition to the previously mentioned XML integration, Adobe has steadily increased helpful features such as tagged PDFs (which reflows text in PDFs to suit different devices), additional commenting options, and more advanced digital signature capabilities. Yet, Hoffman is not alone in this view. In *The Ecology of Computer Viruses*, Jamais Cascio argues that “The relentless spread of a single platform, steadily incorporating more and more interrelated ‘features,’ marginalizes, pushes out and finally kills its ecological competition -- in turn creating the very monocultures that leave the software vulnerable to subversion.” (Cascio, 1999)

Both Hoffman’s and Cascio’s comments dealt with a different monoculture: Microsoft. Yet, much of their views could hold true for Adobe’s PDF specification, as well. The more universal PDF becomes, the more appealing it becomes as a potential target for malicious code. Think about current file types used as delivery vehicles for viruses and worms: Microsoft Word files have served as carriers by way of Word macroviruses; email viruses / worms have propagated largely by using Outlook or Outlook Express as delivery mechanisms. Additionally, so-called “social engineering” virus hoaxes have targeted Windows-based

systems. These virus hoaxes are propagated by way of seemingly helpful email messages warning users to search for and delete specific files on their systems, purportedly containing viruses, but which are actually essential files needed to keep the system running. In each case, the delivery mechanism or catalyst somehow involves a Microsoft product. It is not likely due to Microsoft’s products being less secure than, say, a Mac OS X system. Rather, the Microsoft products present a larger target virtually guaranteeing the virus will spread quickly. As such, the vast majority of viruses and worms actively affect only Windows-based systems. Yet, if PDF were somehow eventually used to propagate viruses, the effect could be significantly more troubling, and could spread significantly more quickly: every Windows-based system, Mac OS X-based system, Linux-based system, and nearly every PDA could potentially be a carrier. If this were to occur, the primary selling point of PDF –universality– could well become the primary problem.

Such a scenario recently surfaced in the PDF monoculture. In August 2001, a new virus known as “Peachy” used PDF files as carriers. The Peachy virus used Acrobat’s well-documented attachment feature, which allows embedding of files within PDF, to spread malicious code. Peachy was a rather tame virus, particularly when compared to newer, highly virulent worms and viruses such as Sobig, Mydoom, and Dumaruru. However, regardless of the mild outcome of Peachy, it caused serious concern. In *New Virus Travels in PDF Files*, author Stephen Shankland states that “the Peachy virus raises the issue that PDF files--widely used to display documents within Web browsers and e-mail--could become a new channel for spreading viruses.” (Shankland, 2001) Similar concerns have been voiced by others. Richard Smith, Chief Technology Officer of the Privacy Foundation, expressed alarm that this event might not be isolated, and that it could in fact be a “new frontier” in virus distribution. (Shankland, 2001)

In April 2003, another PDF-based virus surfaced. The W32.Yourde virus infected files opened and saved in the full version of Acrobat, by exploiting a vulnerability in the Javascript parser of Acrobat. While the Symantec Antivirus Research Center (SARC) rated the Yourde virus as being a “low” cause for concern, and the malicious code only affected Windows-based systems, the end result is still troubling: the PDF was used as an efficient mechanism for virus distribution.

Luckily, both viruses required the full version of Acrobat to execute. Those accessing PDF files using the free Acrobat Reader (recently renamed “Adobe Reader”) remained unaffected, as did those Mac OS X users of Preview. Additionally, most antivirus products are now capable of scanning PDF files for known viruses. However, not everyone uses antivirus protection of their computers, though they should. As such, the chance for similar events occurring in the near future remains a possibility.

The Competition

It could be argued that the continued success of PDF depends on it being a monoculture. That is, since the whole point of the PDF hinges on its universality, any threat to that diminishes its potential for long-term usefulness. In a sense, this universality parallels Metcalfe’s Law: the value of a network increases exponentially with the number of nodes. In this case, the value of PDF as a universal document exchange format increases with the number of people using it. Conversely, the value decreases if people gradually stop using it. Many companies are orchestrating that precise scenario right now, and are pushing their own respective file formats as “the” file format for industry-specific universal document exchange.

Autodesk, for example, has actively campaigned for its Design Web Format (DWF) to displace PDF as the standard for universal document exchange within the architecture and mechanical design industries. Billing DWF as a “simple, secure way to share design data,” Autodesk promotes the DWF files as being

“considerably smaller than PDF and easier to share with downstream users” and adds that “when it comes to distributing design data, nothing beats DWF – not even PDF.” (Autodesk, 2003) Autodesk claims DWF files can be created and accessed quicker than PDF files, which in turn would lead to more convenient and efficient review times for projects. On their web site, Autodesk even includes customer testimonials that openly disparage the PDF, claiming it is ill-suited for mechanical design. (Autodesk, 2003)

In the electronic publishing industry, Microsoft has been pushing its own proprietary eBook file format based on its Reader application. Microsoft Reader has been squarely positioned against the PDF for secure / encrypted eBook content ranging from best sellers to textbooks, and has virtually displaced previous eBook file formats based on the Gemstar eBook file format. (Cesarini, 2002) Amazon now only sells eBooks in formats from Microsoft Reader and Adobe Reader (which uses a PDF specification employing digital rights management for commercial titles).

The problem is two-fold: not only has Microsoft Reader reduced the value of PDF in the eBook industry, it is has rendered certain commercial eBook titles platform-specific by default, since Reader is not available for Mac OS-based systems. This issue is compounded by a third eBook standard competing for market and mindshare, based on Palm Reader. Cora Nucci captured this problem well in *E-Book Dilemma: Potboiler for the Digital Age*. Nucci states:

Imagine having to decide which CD player to buy based not only on features and price, but on what music you’d be able to play on it. Music labels would publish music in one or more formats and pay commissions to the CD player manufacturers based on the sales of each CD issued in its format. If you didn’t own the device(s) that supported discs by your favorite artists, you’d be out of luck. This is pretty much how

the e-book business operates today.
(Nucci, 2002)

Quan echoes this statement in *E-books Face Standards, Copy Protection Issues*. She states that “The lack of a standard file format is a barrier to market growth, since publishers currently converting printed texts must undergo the time-consuming and costly process of accommodating divergent file formats for different reader systems.” (Quan 2000). Admittedly, the eBook industry has been and likely will continue to be in a state of flux for some time until it arrives at a standard eBook file format. However, the fact remains that while PDF is one of the primary file formats in this industry and in mechanical design, it is not the only one, and is universal only as long as competing file formats stay below it. PDF is even being challenged in the web publishing industry by Macromedia’s recent FlashPaper product. FlashPaper is designed to “convert any document into a Flash file, that can be displayed in and printed from a browser window” in a manner purportedly faster and more accurately than PDF. (Becker, 2003)

Un-optimized for Computer Screen and Online Viewing

Usability expert Jakob Nielsen recently voiced serious concerns over the use and apparent misuse of PDF online. In *PDF: Unfit for Human Consumption*, Nielsen states that “Users get lost inside PDF files, which are typically big, linear text blobs that are optimized for print and unpleasant to read and navigate online. PDF is good for printing, but that’s it.” (Nielsen, 2003)

Nielsen refers to numerous problems associated with PDFs, including what he refers to as a host of “usability crimes.” Nielsen argues that most PDF files are usually generated from documents intended for hardcopy. As a result, a great deal of PDF content online does not follow typical guidelines associated with online writing. These files are usually too long, too static, and in the words of Nielsen, often “unpleasant and boring to read.”

He also cites the “jarring user experience” when accessing PDF files online, since these files will either download and open in a separate application, or will launch within the web browser, but with PDF-specific toolbars, buttons, and menu options. Nielsen argues that “even when a PDF file has its own navigation aides, they don't typically help because they're nonstandard and based on a paper metaphor rather than hypertext navigation.” Simply put, PDF files online do not behave similarly to most other online content. Additionally, Nielsen stresses that online PDF files break the flow of reading, since a separate application such as Adobe Reader or Preview must then launch, and PDF layouts are usually optimized for vertical, letter-sized paper, rather than the horizontal dimensions of most desktop and laptop screens. Pivot monitors aside, a great deal of scrolling must be done.

Regardless of whether one agrees or disagrees with Nielsen, he does bring some challenging issues to light. However, these issues are not easily addressed. Fixing them would represent systemic changes to how Acrobat itself functions. Further, since the goal of PDF is to produce an exact replica of a file in a universal format, attempting to somehow interpret the file differently for online use defeats the point of creating a PDF. In view of this, Nielsen suggests “gateway pages” are the least bad solution. That is, rather than directly linking to a PDF, he suggests web designers instead create pages “that summarize the contents of big documents and guide users gently into the PDF morass.” (Nielsen, 2003)

Conclusion

Five years from now, Adobe will still have numerous Acrobat products and the PDF will likely still be robust and widely-used. The strengths Cheng highlights in the current version of Acrobat: simplicity and flexibility, will likely continue to be strong selling points. (Cheng, 2002, p. 63) Aside from versions of the product geared toward specific industries, the overall goal will likely remain the same: ease

document exchange by minimizing the Babylon of competing file formats.

However, it remains unclear if Acrobat will still be the default standard for document exchange. The on-going push for increased XML integration into the PDF standard is definitely a smart move, and can only help make the PDF an even more widely diffused file format in a variety of industries. Yet, if challengers to the Acrobat monoculture succeed only partly, the “ripple effects” felt by these industries could be substantial. Rather than being a universal document exchange format, PDF could be relegated back to being an industry-specific standard. Graphic designers might use it, while mechanical designers use something else, while enterprise users rely on a different standard, and web content producers rely on yet another. Secure, encrypted eBooks may still be in a state of format flux, relying on a combination of PDF and other formats for commercial content. Even worse, standard file formats may become not only industry-specific, but platform-specific, as well.

The road ahead for Adobe's Acrobat product line is unclear and beset with potential problems. If PDF is to remain a “positive monoculture” in document exchange, Adobe must continue to broaden its appeal and increase its feature set and overall usability, yet not water it down so much that PDF becomes the least worst solution, the lowest common denominator in document exchange. Otherwise, it would be Babel all over again.

References

- Adobe. NA. (2005). What is Adobe PDF? *Adobe.com*. Retrieved January 16, 2005, <http://www.adobe.com/products/acrobat/adobepdf.html>
- Adobe. NA. (2004). Acrobat Elements. *Adobe.com*. Retrieved October 5, 2004, <http://www.adobe.com/products/acrobatel/main.html>
- Adobe. NA. (2004) Secure Electronic Documents Drive Efficient Business Processes. *Adobe.com*. Retrieved November 22, 2004, <http://www.adobe.com/products/server/>

- pdfs/95003796_security_solbr_ue2.pdf
- Adobe. NA. (2003). Developer Solutions – Legal Notices. *Adobe.com*. Retrieved December 15, 2003, <http://partners.adobe.com/asn/developer/legalnotices.jsp> [Note: This is a password-protected site]
- Adobe. NA. (2003). Adobe Enterprise Solutions: The Adobe XML Architecture. Retrieved December 18, 2003, from <http://www.adobe.com/enterprise/xml.html>
- Angelo, M. (2003, November 21). Openoffice 1.1 – A Complete Office Productivity Suite. *Mozilla Magazine*. Retrieved December 3, 2003, from http://www.mozillaquest.com/Linux03/OpenOffice_1-1_Overview_Story02.html#Microsoft-Formats
- Apple. (2004). Mac OS X – Architecture. *Apple Computer*. Retrieved September 29, 2004, from <http://www.apple.com/macosx/architecture/>
- Autodesk. (2003). DWF vs. PDF. *Autodesk.com*. Retrieved December 18, 2003, from <http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=3523485>
- Becker, D. (2003, September 30). Adobe's PDF-everywhere Strategy. *News.com*. Retrieved November 8, 2003, from <http://news.com.com/2008-1012-5083805.html>
- Becker, D. (2003, September 18). Adobe e-doc format under siege. *News.com*. Retrieved November 8, 2003, from <http://news.com.com/2100-1012-5078712.html>
- Becker, D. (2003, November 10). Adobe Buys XML Software maker. Retrieved November 12, 2003, from http://news.com.com/Adobe+buys+XML+software+maker/2100-1012_3-5104919.html
- Cesarini, P. (2002). eBooks: A Battle of Standards. *The Writing Instructor*. Retrieved July 12, 2004, from <http://www.writinginstructor.com/essays/cesarini/index.html>
- Cheng, W. (2002). Portable Document Format (PDF)–Finally, a Universal Document Exchange Technology. *Journal of Technology Studies*. 28(1), 59 – 63.

- DuCharm, Robert. (1999). *XML: The Annotated Specification*. Upper Saddle River, NJ: Prentice Hall.
- Evers, J. (2003, July 15). Adobe Pairs PDF, XML. Retrieved July 30, 2003, from <http://www.pcworld.com/news/article/0,aid,111577,00.asp>
- Hoffman, R. (2001). Resisting the Path of Least Resistance. *Network Computing*. Retrieved September 19, 2003, from <http://www.networkcomputing.com/shared/print-Article.jhtml?article=/columnists/1208colhoffman.html&pub=nwc>
- Nielson, J. (2003). PDF: Unfit for Human Consumption. *Useit.com*. Retrieved October 22, 2003, from <http://www.useit.com/alert-box/20030714.html>
- Nucci, C. (2001, February 5). E-Book Dilemma: Potboiler for the Digital Age. *TechWeb.com*. Retrieved July 22, 2002, from <http://content.techweb.com/wire/story/TWB20010205S0002>
- Olavsrud, T. and Orzech, D. (2003, July 15). Adobe Ties PDF Format to XML. Retrieved July 20, 2003, from <http://www.cioupdate.com/news/article.php/2235401>
- Quan, M. (2000, September 28). E-Books Face Standards, Copy Protection Issues. *TecWeb.com*. Retrieved November 14, 2002, from <http://content.techweb.com/wire/story/TWB20000928S0004>
- Quigley, P. (2003, April 22). Adobe Pulls XML into PDF Mix. Retrieved June 6, 2003, from <http://www.contentmanagement365.com/Content/articles/article924.aspx>
- Shankland, S. (2001, August 7). New Virus Travels in PDF Files. *News.com*. Retrieved December 22, 2003, from <http://news.com.com/2100-1001-271267.html?legacy=cnet>
- Symantec Antivirus Research Center. (2003, May 2). W32.Yourde. *SARC.com*. Retrieved July 3, 2003, from <http://securityresponse.symantec.com/avcenter/venc/data/w32.yourde.html>
-