



# The Structure of PDF, or: whither Acrobat?

**Thomas Merz**

**PDFlib GmbH  
München, Germany  
[www.pdfliib.com](http://www.pdfliib.com)  
[tm@pdfliib.com](mailto:tm@pdfliib.com)**



# 1980s: Building the Foundations

- ▶ PostScript sparks the desktop publishing revolution
- ▶ While PostScript accounts for the vast majority of printed material, it's not very much suited to online and interactive use
- ▶ Adobe Illustrator 88 serves as a test bed for using PostScript as editable file format
- ▶ The AI file format will later form the basis for PDF



## 1991-1993: The Story begins

- ▶ John Warnock: »Revisability will come«, PostScript and SGML will grow together
- ▶ Acrobat 1.0 (code named »Carousel«) launched in summer of 1993 (best of Comdex award fall 92)
- ▶ Acrobat Distiller 1.0 support PostScript Level 2
- ▶ pdfmark operator for hypertext elements in PostScript already supported
- ▶ Acrobat Reader US-\$ 50
- ▶ PDF 1.0 Reference Manual published by Addison-Wesley



## 1994: Early Adopters

- ▶ **September 1994: Acrobat 2.0 (with PDF 1.1) launched**
- ▶ **Acrobat supports the Plugin interface; improved developer support**
- ▶ **Adobe Developers Association (ADA) ships disks with technical documentation in PDF**
- ▶ **U.S. tax authorities ship CD-ROMs with administrative documents in PDF**
- ▶ **Acrobat supports encryption, search, and weblink**
- ▶ **Acrobat Distiller sold separately from Exchange**
- ▶ **Acrobat Reader freely available**
- ▶ **Acrobat 2.0 adopted as standard for AdSend**
- ▶ **Bundling agreement with CompuServe**
- ▶ **Early Acrobat competitors: Common Ground, Replica**



# 1995: Application support, and User Adoption

- ▶ **Capture 1.0 US-\$ 4000**
  - English only
  - no European special characters
  - black and white only
- ▶ **FrameMaker 5.0 supports Acrobat 2**
- ▶ **People start using Acrobat for review and annotation purposes**
- ▶ **PageMaker 6 includes Acrobat 2**
- ▶ **Prepress use of Acrobat:**
  - PDF not only for exchange of screen documents, but for shipping data to the prepress service bureau
  - color separation missing in Acrobat
  - big advantages through built-in compression



## 1996: Adobe discovers the Web

- ▶ **PDF/Internet integration:**
  - PDF documents appear on the Web
  - linking between HTML and PDF
- ▶ **Acrobat 3.0, code named »Amber«**
  - single package US-\$ 300
  - Internet integration: WebLink and page-at-a-time download (kludge)
  - downloading individual pages of a PDF is not possible
  - free Netscape plugin for PDF viewing
  - forms
  - TouchUp tool
  - improved prepress support: OPI, CMS, screening
  - plugin interface adopted by many software vendors



# 1997: No longer a Niche Market

- ▶ Acrobat plugin market expands:
  - PitStop for editing PDF
  - automatic bookmark and link generation
  - database and catalogue publishing
  - many, many more
- ▶ First PDFlib versions available:
  - dynamic PDF on the server
  - non-interactive PDF generation
- ▶ CGATS (Graphic Arts Technical Committee adopts PDF as a standard for transmitting digital ads
- ▶ Seybold Seminars include a full day dedicated to Acrobat and PDF
- ▶ Application support for PDF is still far from perfect:
  - MS Office users are left out in the cold
  - »pdfmark Primer« downloads nearly break my Web site



## 1998: Getting serious

- ▶ PageMaker plugin for importing PDF:
  - great concept, but limited practical use
- ▶ German and Swiss experts publish whitepaper on PDF for prepress:
  - need Distiller profiles to facilitate option handling
  - PDF import facility needed in major DTP applications
  - composite PDF and trapping don't work due to Quark XPress problems
  - cropping, bleed, crop marks are missing
  - severe page size constraints limit PDF imposition and ad distribution
  - spot color handling is only rudimentary possible
  - font embedding problems
  - no separation preview
- ▶ Crucial Acrobat Plugins for prepress work:
  - Enfocus CheckUp checks all relevant properties of a PDF file (preflight)
  - Quite Imposing delivers affordable imposing with Acrobat
  - Lantana Crackerjack provides color separation (leverages in-RIP separation)
  - Enfocus PitStop for editing PDF page contents



## 1998: Waiting for the next Version

- ▶ Agfa delivers first PDF-based workflow system Apogee
- ▶ Security plugins protect intellectual property:
  - companies start selling PDF on the Web
  - protect business assets through encryption
- ▶ Third-party products fill some functional gaps in Acrobat:
  - annotation plugins
  - editing plugins
  - productivity tools for dealing with large amounts of PDF documents
  - batch processing



## 1999: Acrobat 4 arrives

- ▶ Acrobat Exchange, a program contained in the Acrobat package, is now called »Acrobat« in order to avoid confusion...
- ▶ Business and office use:
  - Microsoft Office support
  - drag-and-drop PDF generation for the novice user
  - improved annotation and markup features
  - digital signatures and compare documents
- ▶ Prepress:
  - Distiller profiles
  - high-end PDF: color profiles, spot color, blends
  - Prepress use still limited due to a number of severe bugs
- ▶ Internet integration: Web capture makes Acrobat a Web browser, and preserves HTML contents as PDF
- ▶ Microsoft Australia inadvertently post a PDF file on their Web server (discovered by Karl de Abrew)



# 1999: Internationalization

- ▶ Acrobat 4 ships in a variety of localized version
- ▶ Unicode support in the file format and the software:
  - Unicode is easily available for hypertext elements
  - PostScript font heritage still slows down full Unicode support for page descriptions
- ▶ CID font support for Chinese/Japanese/Korean languages
  - Adobe freely distributes a set of CJK fonts
  - non-CJK Acrobat versions display and print Asian documents



## 2000: The Real World

- ▶ **Problems with Acrobat 4.0:**
  - severe bugs in prepress use
  - several Macintosh plugins missing
  - Adobe Acrobat Distiller doesn't like Adobe FrameMaker's PostScript output when Adobe printer driver is used...
- ▶ **Acrobat 4.05 addresses many issues but:**
  - very long delays in delivery
  - my German copy of Acrobat 4.05 arrived in May 2000 – six months after the announcement, and after many phone calls
  - Acrobat 4.05a updates Acrobat 4.05 but you don't necessarily need it?
  - confusion caused by commercial terms, availability, and numbering



## 2000: PDF in Prepress

- ▶ Several high-end workflow solutions available:
  - Agfa Apogee
  - Heidelberg/Creo Prinergy
  - Scitex Brisque
- ▶ Scores of plugins available
- ▶ Variable data printing: PPML wraps PDF and other formats
- ▶ Evolving PDF/X standard for prepress data exchange
- ▶ Yet another attempt at standard job tickets: Job Definition Format (JDF)
- ▶ Editability of PDF?



## 2000: The impact of PDF on the Web

- ▶ PDF usage on the Internet is beyond the critical mass:
  - 100 million (?) copies of Acrobat Reader downloaded over the Web
  - according to StatMarket, 36% of all Netscape users have the Acrobat plugin installed (rank 5 after AVI, QuickTime, Beatnik, RealPlayer plugins)
  - corresponding number of Acrobat ActiveX use in MS Internet Explorer is hard to find
- ▶ Number of search hits on AltaVista (only HTML documents queried!):
  - HTML: 12 million
  - PDF: 4 million
  - SGML: 275 000
  - XML: 665 000
  - PostScript: 524 000



## 2000: PDF on the Server Side

- ▶ Increasing demand for server-side PDF generation
- ▶ Acrobat Distiller server for Intranet or Internet use announced
- ▶ XML-based publishing systems:
  - document handling and manipulation in XML
  - user presentation (low-end) in HTML, or XML plus style sheets
  - user presentation (high-end) in PDF
- ▶ PDFlib generates PDF on the server side:
  - easy-to-use API
  - dozens of platforms supported
  - a variety of programming languages and Web environments supported
  - fast PDF generation
  - for more information see <http://www.pdfliib.com>



## 2000: Alternative Clients

- ▶ **Acrobat Viewer for Java:**
  - platform-independent
  - exciting technology, but resource hog
  - limited in functionality, plugins not available
- ▶ **PDF on PDAs:**
  - Ansyrr Primer for Windows CE; versions for Pocket PC and Palm OS announced
  - does PDF's page paradigm match palm-size usage?
- ▶ **Adobe Document Server (ADS) converts PDF to HTML and raster-based image formats for viewing on non-PDF aware clients**



## 2000: PDF and the E-Thing

- ▶ Adobe uses the term ePaper for...?
- ▶ Acrobat Web-Buy and PDF Merchant: sell encrypted PDF over the Web
  - purchased PDFs are bound to a particular machine, or disk, or operating system
- ▶ E-Books with PDF:
  - Glassbook Reader: Acrobat Reader, packaged with a book protection and distribution system
  - EveryBook Dedicated Reader: hardware device with built-in PDF support and double-screen
- ▶ Evolving E-Book standards:
  - Open eBook (OEB): document format based on XML and CSS
  - Electronic Book Exchange (EBX): not a document format but a cryptographic distribution system; can be used with PDF



## 2000: PDF looks good, but...

- ▶ Repurposing PDF content is still difficult:
  - While perfectly preserving the layout, Acrobat still doesn't »know« much about the document's contents
  - pages, columns, lines – it's all just characters placed somewhere
  - different document views (screen vs. print)
  - PDF guarantees the appearance, but ignores the structure
- ▶ Users wish to...
  - cut-and-paste regardless of creation platform, font, and encoding
  - extract text so that it can be re-used (e.g., remove hyphenation, re-assemble connected text columns)
  - round-trip between PDF and other formats without losing information
- ▶ Third-party tools are good at conversion and extraction, but must work heuristically



## 2000: XML hype all over the world

- ▶ XML (Extensible Markup Language) is a meta-language for describing documents and data of any kind; »looks« like HTML because of its tags
- ▶ XML is derived from SGML, but offers 90% of SGML's functionality with 10% of its complexity
- ▶ Advantages of XML over SGML:
  - lower complexity
  - extensible
  - web-aware
- ▶ Strict division of content structure and appearance:
  - structure is described by markup (similar to HTML tags)
  - appearance is described by style sheets
- ▶ XML is targeted as a universal platform for data exchange



## 2000: Sidebar—going from XML to PDF

- ▶ XML has all the document structure, but no formatting
- ▶ Style sheets add the required formatting
- ▶ Extensible Stylesheet Language (XSL) will be the W3C-approved standard:
  - XLST for transforming XML documents
  - formatting objects (FO) for describing layout semantics
- ▶ A generic formatting engine is required:
  - take XML and apply the stylesheet
  - output the result as PDF
- ▶ The XEP processor does just that:
  - fully Java-based XSL/FO rendering engine for XML to PDF conversion
  - uses PDFlib as PDF backend
  - for more details see <http://www.renderx.com>



## 2000: Structure Features of Acrobat 4/PDF 1.3

- ▶ PDF supports embedded structure information inside the document
- ▶ Huge potential, but rarely used in today's applications
- ▶ Can make the old promise true and combine PostScript and SGML (or PDF and XML, or document appearance and structure)
- ▶ Finally: repurpose your document contents, while keeping them in final format
- ▶ Archive both appearance and structure in a single file format
- ▶ Possible applications:
  - extract contents
  - intelligently convert PDF to other formats
  - aid document navigation
  - round-tripping



## 2000: Structured PDF—some Details

- ▶ We need data *about* documents in addition to the actual contents:
  - meta data = information about the document as a whole
  - structure = information about portions of the document, and their relationships
- ▶ Basic problem: text in PDF is often heavily fragmented; how to address individual content portions?
- ▶ Identify contents in the document:
  - attach text blocks to logical elements
  - contrary to XML, not only text can be part of an element, but any kind of rendered content (marked content, XObjects)
- ▶ Content is organized in a tree-like structure:
  - hierarchical system of logical elements
  - each element points to some document content, or other elements
- ▶ Predefined standard elements for logical units (similar to HTML)
- ▶ Additional application-specific logical elements can be defined



## 2000: PDF Element Types and Attributes

- ▶ The element type may be regarded as the »tag name« of the element
- ▶ Adobe suggests the use of standard types, similar to HTML tags
- ▶ Standard elements for linearly ordered text in a hierarchy: Heading, Section, Paragraph, List, ListItem, Caption, Table, TableHeader, TableData, Figure, Index, TableOfContents...
- ▶ Standard element types could help in repurposing and round-tripping
- ▶ Application-specific elements, e.g., order number, invoice item
- ▶ Attributes further characterize structure elements, and define the application-specific role of an element:
  - standard attributes: link, reference
  - attributes »belong« to the application which created them



## 2000: Generating structured PDF

- ▶ **Acrobat Web Capture:**
  - translates structure information from HTML document to PDF structure
  - the document has some knowledge about its origin
- ▶ **PDFMaker for Microsoft Word:**
  - generates structured PDF from paragraph styles
- ▶ **FrameMaker 6.0:**
  - generates structured PDF from paragraph styles
- ▶ **Adobe Illustrator and InDesign already use structure and ClassMap privately**
- ▶ **Technical approaches:**
  - pdfmark operator: PostScript code contains structure information
  - Acrobat's plugin programming interface



## 2000: Using structured PDF

- ▶ **Structured Bookmarks:**
  - contrary to common bookmarks, structured bookmarks don't point to a specific location in the document but to a specific structure element
  - structured bookmarks may be attached to a page range
  - they »know« about the corresponding document parts
- ▶ **Structured bookmarks can use the information which is attached to them:**
  - print all pages of this bookmark
  - delete structured bookmark and all corresponding pages
  - extract pages
  - move bookmarks and corresponding pages (ctrl-drag bookmark)
- ▶ **Generate bookmarks based on the structure information**
  - »New bookmarks from structure«
- ▶ **Structured bookmarks with attached Web information:**
  - »Append Next Level«
  - »View Web Links«
  - »Open Page in Web Browser«



## 2000: Example – the WINDS Newspaper Project

- ▶ **WINDS (Worldwide Instant Newspaper Distribution Services) is an international project, financed by the European Community**
- ▶ **Participating companies include Adobe as technical leader, and several newspaper publishers**
- ▶ **Distribute newspapers in PDF via Internet and satellite broadcast**
- ▶ **Repurpose the printed edition for e-commerce**
- ▶ **WINDS PDF newspaper contains page appearance plus logical structure**
- ▶ **The NewsReader Acrobat plugin uses the structure information:**
  - reformat lengthy articles in a more screen-friendly format
  - create individual news digests by extracting selected articles
  - similar techniques on the server side
- ▶ **Commercial implementation uses PDF Merchant and WebBuy**
- ▶ **More information available at <http://www.winds-eu.com>**



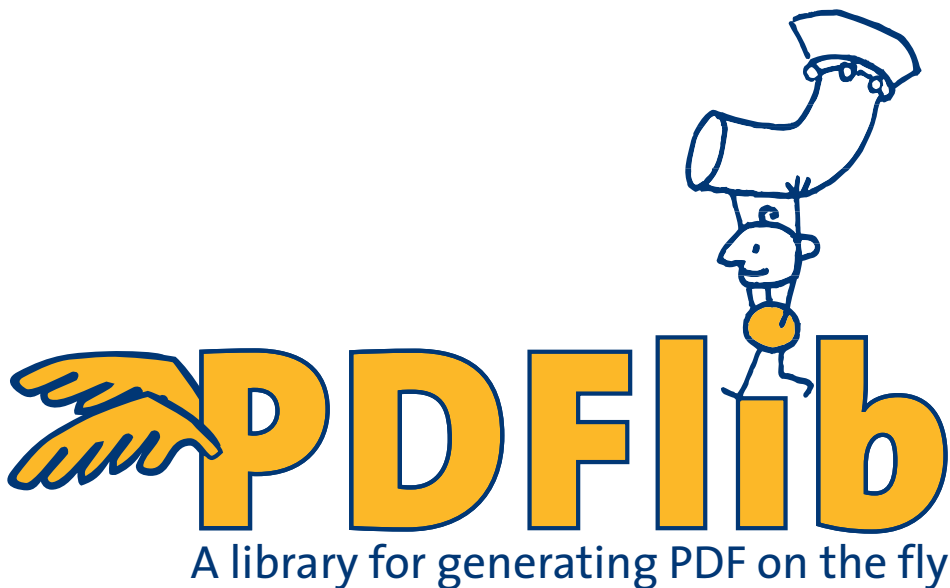
## 2001: Improved Application Support for PDF

- ▶ Applications fully support PDF export:
  - PDF export with full Distiller flexibility, but without the need for PostScript
  - document structure is preserved on PDF export
- ▶ Applications fully support PDF import:
  - PDF import with full editability
  - document structure is retained on PDF import
- ▶ New quality of PDF usage on the Web server:
  - XML-based publishing systems shuffle around data
  - On-demand PDF generation if the user requests it
  - PDF includes structure information for easier reuse
- ▶ Microsoft still tries to ignore PDF...?



## 2001: Leveraging Structure Information in PDF

- ▶ **New Acrobat features (?):**
  - save as XML
  - extract all columns comprising an article
  - jump to chapter 9 (without any link or bookmark in place)
  - print this chapter and the next one
  - context-sensitive search: »search all captions for this text«
  - chapter and section numbers displayed in search result list
  - link management: embedded PDF links are adjusted automatically when the text or page is changed
  - navigate the document in a tree-like structure



### ***What is PDFlib?***

Want to spice up your generated documents with PDF? Tired of HTML formatting issues? PDFlib is a development tool for PDF-enabling your software, generating PDF on your server, or distributing dynamic PDF content over the Web. PDFlib frees you from the intricate details of PDF generation by offering a simple API for programmatically creating PDF files from within your own server- or client-side software.

PDFlib doesn't require Adobe Acrobat nor any other third-party software.

### ***PDFlib everywhere!***

PDFlib runs on a wide variety of platforms: Mac, Windows, and many Unix platforms are supported. In addition, PDFlib's text and image handling has been carefully crafted to accommodate EBCDIC-based platforms such as the IBM AS/400.

The PDFlib core is written in the ANSI C language. In addition, we undertook great efforts to make the PDFlib API accessible from a variety of other programming environments by leveraging the extension mechanisms offered by most modern languages. The integrated support for several scripting languages makes PDFlib especially attractive for devel-

opers who want to concentrate on their actual problem instead of the programming environment. PDFlib 3.0 supports the following language bindings:

- ▶ ActiveX/COM for use with Visual Basic, Active Server Pages, etc.
- ▶ ANSI C
- ▶ Class wrapper for ANSI C++
- ▶ Java (JDK 1.1.x or 1.2.x)
- ▶ Perl
- ▶ Python
- ▶ Tcl

## ***API***

PDFlib offers an easy-to-use programming interface for the application programmer. The PDFlib API shields the programmer from the technicalities of PDF generation. Any programmer with decent graphics or print output experience is able to use PDFlib quickly, and will be able to incorporate PDFlib into his application within a couple of hours. The PDFlib reference manual explains the basics of PDFlib programming, and provides a detailed reference to all API functions. Sample programs are provided for all supported environments.

## ***Web Server Deployment***

PDFlib is thread-safe, i.e. it can safely be used in multi-threaded server applications. The ActiveX edition is »both-threaded« for improved performance. C or C++ library clients can install their own memory management and error handling routines. PDFlib's memory management has been rigorously engineered and tested for memory leaks in order to guarantee 24 x 7 deployment without any shutdown periods.

PDFlib is especially well suited for generating PDF on the Web server, and is tightly integrated into all major Web server environments. PDFlib can generate PDF data directly in memory (instead of on file), resulting in better performance and avoiding the need for temporary files.

## ***PDF Features supported by PDFlib***

### ***PDF Documents***

- ▶ PDF documents of arbitrary length, directly in memory (for Web servers) or on disk file
- ▶ Arbitrary page size—each page may have a different size
- ▶ Compression for text, vector graphics, image data, and attachments
- ▶ Strict Acrobat 3 / PDF 1.2 mode optionally available

### ***Vector graphics***

- ▶ Common vector graphics primitives: lines, curves, arcs, rectangles, etc.
- ▶ Vector paths for stroking, filling, and clipping
- ▶ RGB color for stroking and filling objects

### ***Fonts***

- ▶ Text output in different fonts
- ▶ Text column formatting
- ▶ Underlined, overlined, and strikeout text
- ▶ Built-in font metrics for PDF's 14 base fonts
- ▶ PostScript font embedding (PFB and PFA file formats)
- ▶ Support for AFM and PFM font metrics files
- ▶ Library clients can retrieve character metrics for exact formatting
- ▶ Flexible font and metrics file configuration

### ***Hypertext***

- ▶ Page transition effects, such as shades and mosaic
- ▶ Nested bookmarks
- ▶ PDF links, launch links (other document types), and Web links
- ▶ Document information: four standard fields (Title, Subject, Author, Keywords) plus user-defined info field (e.g., part number)
- ▶ File attachments and note annotations

### ***Internationalization***

- ▶ Unicode support (see below)
- ▶ Support for a variety of encodings (both built-in and user-defined)
- ▶ CID font and CMap support for Chinese, Japanese, and Korean text
- ▶ Support for the Euro character
- ▶ Support for international standards, e.g., ISO 8859-2

### ***Images***

- ▶ Embed images in GIF, PNG, TIFF (single- and multi-page), JPEG, and CCITT file formats
- ▶ Image data can be passed directly in memory
- ▶ Efficiently re-use image data, e.g., for repeated logos on each page
- ▶ Transparent (masked) images

## ***Support for the Unicode Standard***

PDFlib supports the Unicode standard as far as PDF itself supports it. Unicode text can be used for hypertext features such as bookmarks (e.g., Greek or Russian), contents and title of text annotations, attachment description and author name. In addition, Unicode encoding (CMaps) can be used for Japanese, Chinese, and Korean text. PDFlib's Unicode handling is available for all supported client language bindings, and is transparently integrated in those language bindings which natively support Unicode themselves. Currently these are ActiveX, Java, and Tcl.



## ***Who uses it?***

Since PDFlib's inception in 1997, thousands of software developers have downloaded PDFlib over the Internet. PDFlib is very popular among Web developers, developers of financial and reporting software, in-house applications for banks and insurances, and end-user software. A list of reference customers can be found on our Web site.

## ***License Fees***

We offer two different licensing schemes: server licenses (mostly for Web use) and redistributable (runtime) licenses for product developers. All licenses include six months of technical support. Details of one-year maintenance contracts and OEM licenses (incl. source code license) are available on request.

<i>server license</i>		<i>redistributable (runtime) license</i>	
<i>server CPUs</i>	<i>license fee</i>	<i>units distributed</i>	<i>license fee</i>
<i>1-2</i>	<i>US-\$ 500</i>	<i>1-99</i>	<i>US-\$ 500</i>
<i>3-16</i>	<i>US-\$ 1000</i>	<i>100-999</i>	<i>US-\$ 1000</i>
<i>unlimited</i>	<i>US-\$ 2000</i>	<i>unlimited</i>	<i>US-\$ 4000</i>

## ***Contact***

Fully-functional versions of PDFlib can be obtained from our Web site. For further information regarding PDFlib licensing please contact:

PDFlib GmbH, Tal 40, 80331 München, Germany  
phone +49 • 89 • 29 16 46 87 sales@pdflib.com support@pdflib.com  
fax +49 • 89 • 29 16 46 86 www.pdflib.com