Portable Document Format

目次 非表示
ページ先頭
概要
PDFの特徴
PDFの利用場面
PDFの歴史
技術的な基盤
技術概要
技術の特筆点
ソフトウェア
PDFの短所
脚注
関連項目
外部リンク

出典: フリー百科事典『ウィキペディア（Wikipedia）』

「**PDF**」はこの項目へ転送されています。その他の用法については「PDF (曖昧さ回避)」をご覧ください。

ウィキペディアにおけるページをPDF形式でダウンロードする方法については、「Help:PDF形式でダウンロードする」をご覧ください。

この記事は言葉を濁した曖昧な記述になっています。Wikipedia:言葉を濁さないおよびWikipedia:避けたい言葉を参考に修正してください。（2019年5月）

Portable Document Format（ポータブル・ドキュメント・フォーマット、PDF）は、デジタルデバイス上でアプリケーションやOS、ハードウェアに依存せず文章や図版を表示するために開発され、ISO 32000[2]で国際標準化された電子文書ファイル形式である。 PostScriptをベースにAdobeが開発し、1993年にAdobe Acrobatで初めて採用された。

Portable Document Format

拡張子	.pdf
MIMEタイプ	application/pdf[1]
タイプコード	'PDF '（半角空白を含む）
UTI	com.adobe.pdf
マジックナンバー	%PDF–
開発者	ISO (2008年–現在)
初版	1993年6月15日 (31年前)
最新版	2.0 (2017年7月28日 (6年前))
種別	ページ記述言語
派生元	PostScript
拡張	PDF/A, PAdES, PDF/E, PDF/UA, PDF/VT, PDF/X
国際標準	ISO 32000-2
オープンフォーマット	はい。
ウェブサイト	www.iso.org/standard/75839.html

概要 [編集]

特定の環境に左右されずに全ての環境でほぼ同様の状態で文章や画像等を閲覧できる特性を持っている。

アドビはPDF仕様を1993年より無償で公開していたが、それでもPDF規格はアドビが策定するプロプライエタリなフォーマットであった。

2008年7月には国際標準化機構によってISO 32000-1として標準化された[3][4][5]。

アドビはISO 32000-1 についての特許を無償で利用できるようにしたが、XFA（英語版）(Adobe XML Forms Architecture) やAdobe JavaScriptなどはアドビのプロプライエタリな部分として残っている。

PDFの特徴 [編集]

PDFファイルは 印刷可能なあらゆる文書から生成でき、表示や印刷は各社が無料、有料で配布しているソフトウェアを利用する。

主な物にアドビが無料で配布しているAcrobat Reader（旧Adobe Reader）がある。

また、AppleのmacOS, iPadOS, iOSではOSレベルでPDFの表示、編集、作成、印刷に対応している[6][7]。

特徴 [編集]

PDFには、次の特徴がある。

- 作成した文書を異なる環境のコンピュータで元のレイアウト通りに表示・印刷できる
- 文書のセキュリティを設定できる
- 圧縮してデータを格納することで、ファイルサイズを小さくできる
- しおり、リンク、注釈といった、文書を画面に表示するときに便利な機能を設定できる
- フォーム機能を使って、利用者の入力欄を受け取るような書式設定済み文書を作成できる
- 音声化などアクセシビリティ に配慮した文書を作成できる
- マルチメディア に対応している

レイアウトの保持 [編集]

PDFは、Acrobat Readerに代表されるPDFビューアー（表示用のソフトウェア）がインストールされているコンピュータであれば元のレイアウトどおりに表示・印刷できる。Acrobat Readerは Windows、macOS、Android、iOS など各種オペレーティングシステム (OS) に対応したものが無償で配布されており、他のPDF閲覧ソフトも数多く存在するため、PDFファイルは多くの環境で閲覧・印刷できる。

PDF以外のテキスト形式の電子文書は、ほかのコンピュータ上で元のレイアウトを保持したまま表示・印刷するのは難しい。例えば、WordやExcelなどMicrosoft Officeの文書は、対応するソフトウェアをインストールすれば閲覧することは可能だが、バージョンや設定が違っていたり、フォントの有無が原因で編集者が意図した通りのレイアウトを保てない場合がある。HTML の文書は多くのコンピュータで閲覧できる。しかし、レイアウトの制限が大きい上、OSやウェブブラウザの種類・設定でレイアウトが変わりやすい。

そのため、厳密にレイアウトを保持する必要のある文書はPDF化することが多い。ただし、フォントの設定によっては、PDFでも元のレイアウトを保持できない場合がある。この問題は、フォントを埋め込むことで回避できる。

PDFの利用場面 [編集]

電子文書の公開・配布 [編集]

PDFの特長は、PDFファイル作成元と異なるコンピュータ環境において、作成元文書のレイアウトや書式を忠実に再現した表示・印刷ができることにある。その性質を好んで、ウェブページ上の文書やソフトウエアの説明書などの広く公開・頒布する形態で多く利用される。また、同様の理由から、将来のコンピュータの環境変化を想定した長期保存向け文書 (PDF/A) や、 DTP の過程でPDFファイルを作成・利用する (PDF/X) 用途も多くなっている。

印刷物として制作した文書のPDF化 [編集]

Quark XPressやAdobe InDesignなどのDTPソフトウェアで組版した結果のデータは、しばしばPDFファイルとして出力される。こうして作成されたPDFファイルは、印刷物と同じレイアウトの電子文書となる。一般に、PDFファイルの公開・配布は印刷物を配布するのに比べて低コストである。

そのためPDFを利用して例えば、カタログやパンフレットなどをインターネット上で公開したり、マニュアルや雑誌の収録記事をCD-ROMで配布することが多くなっている。

PDF入稿 [編集]

印刷物制作時の入稿をPDFですることも増えてきている。従来は、QuarkXPress などで組版した結果のデータをそのまま入稿することが多かった。

PDF入稿には、

- 原稿作成方法の制限が小さくなる
- 画像ファイルの添付し忘れやエラーの発生を少なくできる
- データサイズをコンパクトにすることができる

などといった利点がある。

ただし作成方法によっては、商業印刷には使えないPDFファイルが生成されることもある。たとえば、紙資料をスキャンして作成したPDFファイルから商業印刷に要求される結果を得るのは難しい。目的とする印刷品質を得るためには、フォントの埋め込みや印刷時に使用する色の情報、画像解像度などをPDFファイル作成時に適切に設定する必要がある。この設定を行うにはコンピュータの操作方法ならびに印刷物とその製造工程を的確に理解していることが必須であるため、誰でも確実に行うことができるとは言い難い。PDF/Xは、こうした問題を回避するために用いられる[8]。

データの入力と収集 [編集]

Acrobatは「フォーム」機能を利用し、文書入力のインタフェースとしても利用される。フォームとは記入用紙のことで、PDF内に記入欄を設け、エンドユーザーに入力させてメールで送付させる、あるいはPDFのインタラクティブ機能を使って サーバに送信させることでデータを収集することができる。

またPDFファイルにはJavaScriptを埋め込むことができる。これにより入力のナビゲーションや入力時の値チェックなどが可能になる。

PDFの歴史 [編集]

詳細は「PDFの歴史」を参照

経緯 [編集]

PDFは 1990年代初めに、文書共有を目的として開発された。文書の書式とインラインの図表を保持し、異なるプラットフォームのコンピュータのユーザー間で文書を閲覧するために互換性のあるアプリケーションを利用できない場合にも文書を共有できる[9]。競合としてDjVu (開発続行中)、Envoy、Common Ground[10]社の DigitalPaper, Farallon Replica[11]の他にアドビ自身が推進するPostScript format (.ps) がある。初期、 World Wide WebとHTMLが興隆する以前のPDFの主要な用途はデスクトップパブリッシングのワークフローであった。

PDFの初期の普及の足取りはゆっくりしたものであった[12]。アドビのPDF閲覧・作成ソフトAdobe Acrobatは無償ではなく、また初代バージョンのPDFは文書外部へのハイパーリンク を提供しないためインターネットの特長を活かしていなかった。またプレーンテキストと比べてサイズは大きいためモデムによるダウンロードは時間がかかり、また当時ハイスペックだったPCでも表示は遅かった。

バージョン 2.0より、アドビはAcrobat Readerを無償配布するようになった[13]。初代バージョンとの互換性は保たれており次第に書式を固定した電子文書のデファクトスタンダードの地位を確立した[14]。

2008年にアドビのPDF Reference 1.7はISO 32000:1:2008として策定された。以後のPDF（PDF 2.0を含む）開発はIS のTC 171 SC 2 WG 8においてアドビと専門家らの協力により進められている。

一覧 [編集]

時	出来事
1993年	アドビシステムズ（現アドビ）、PDF1.0とAcrobat 1.0をリリース。
1994年	アドビシステムズ、Acrobat Readerの無償配布開始。この無償配布が、PDF普及の大きな要因となった。
1995年	アドビシステムズ、 Netscape Navigator用のAcrobatプラグインを公開。ウェブブラウザー上でのPDF利用を促進した。
1996年	アドビシステムズ、PDF 1.2とAcrobat 3.0をリリース。このバージョンからPDFとAcrobatが日本語に対応。
1999年	Apple、Mac OS Xをリリース。 Quartzを採用し、OSレベルでPDFへ対応。 アドビシステムズ、PDF 1.3とAcrobat 4.0をリリース。
2001年	アドビシステムズ、PDF 1.4とAcrobat 5.0をリリース。
2003年	アドビシステムズ、PDF 1.5とAcrobat 6.0をリリース。
2004年	ソースネクスト、日本国内で「いきなりPDF」シリーズを発売開始。低価格のPDF作成ソフトということで注目を集めた[注 1]。 アドビシステムズ、PDF 1.6とAcrobat 7.0をリリース。
2005年	アドビシステムズ、 マクロメディア を買収。PDFとFlashの統合が開始された。
2006年	アドビシステムズ、PDF 1.7とAcrobat 8.0をリリース。
2008年7月2日	ISOの管理規格となる。ISO 32000-1。
2008年	アドビシステムズ、PDF 1.7, Adobe Extension Level 3とAcrobat 9.0をリリース。
2010年	アドビシステムズ、PDF 1.7, Adobe Extension Level 8とAcrobat X(10.0)をリリース。

アドビによる仕様定義 [編集]

1993-2006年にかけてアドビはPDFの仕様を数回改訂して新機能を追加している[15][16]。

PDF 1.7 (ISO 32000-1) が制定された2006年以降にアドビが定義した拡張 (Extension Levels) はISO 32000-2 (PDF 2.0) のドラフトにも一部取り込まれているものもあるが、開発者はアドビの拡張部分がPDFの標準でないことに注意する必要がある[17]。

PDFのバージョンと仕様書 [編集]

バージョン	仕様書[5]	発行年	新機能	Acrobat Reader のバージョン
1.0	Portable Document Format Reference Manual[18]	1993	—	Carousel
1.1	First, revised[19]	1996	• 暗号化（MD5, RC4 40ビット） • device-independent color • スレッド • リンク	2.0
1.2	First, revised[19]	1996	• インタラクティブなページ部品（ラジオボタン, チェックボックス等） • インタラクティブな記入フォーム • Forms Data Format (FDF) により記入内容のインポート、エクスポートをインターネット経由で行えるようになった • マウスのイベント • 外部動画の再生 • 外部または添付の音声の再生 • zlib/deflateによるテキストと画像の圧縮 • Unicode対応 • 色管理と代替画像のサポート	3.0
1.3	Second[20][21]	2000	• 日本語フォントの埋め込み • 電子署名 • ICCおよびDeviceN色空間 • JavaScript • 各種ファイルストリームの添付 (ファイル添付) • 注釈種類の追加 • Adobe PostScript Language Level 3 imaging model で追加された機能のサポート • イメージのマスク • 代替イメージ • スムージング • ページ番号付けの強化 • ウェブキャプチャ • 表示順以外に論理構造表現のサポート • CIDフォントのサポート強化 • data structures for mapping strings and numbers to PDF objects • プリプレス分野のワークフローのサポート • new functions for several function object types that represent parameterized classes of functions[20][22] • Acrobat JavaScript Object Specification Version 4.05	4.0
1.4	Third[23]	2001	• JBIG2画像圧縮 • 透明効果 • OpenTypeフォント対応 • RC4暗号化のキー長が40ビットから128ビットまでに拡張 • 入力フォームのインタラクティブ性の強化 (FDF) • XMLフォーム投稿、FDFファイルの添付 • エクスポート時のUnicode対応 • FDFファイルの共同編集と署名の追加 • 障碍者向けアクセシビリティ機能 • Extensible Metadata Platform (XMP) によるメタデータストリーム • タグ付きPDF • inclusion of printer's marks • 編集時のページ境界の表示 • CMaps（フォント対応表）の拡充 • alternate presentations • PDFファイル間のインポート • データ添付のディクショナリのサポート[22][23] • Acrobat JavaScript Object Specification Version 5.1[24]	5.0
1.5	Fourth[25]	2003	• JPEG 2000画像圧縮 • マルチメディアの添付と再生の強化 • object streams • cross reference streams • フォーム入力の XML Forms Data Format (XFDF) 対応（PDF 1.4のXML対応を代替） • フォーム • リッチテキスト • 属性を XML Forms Architecture (XFA) 2.02で記述できるようになった（静的XFAフォームのみ対応） • PKCS#7公開鍵のサポート（PDF 1.3で追加されたが1.5のリファレンスに初めて記載された） • 公開鍵による暗号化 • アクセス許可 • ユーザー権限の署名（本文を暗号化する必要がなくなった） • SHA-1 • RSA鍵の4096ビットサポート	6.0

			• 独自の暗号化、復号を実装できるようになった • 文書の一部を可視、不可視にする （CAD、レイヤー図画、地図、多言語文書などに対応） • JavaScript で実装するスライドショー表示をサポート（Reader は SVG 1.0のみ対応）[22] • Acrobat JavaScript Scripting Reference, Version 6.0[26] • Microsoft Windows 98サポートの廃止	
1.6	Fifth[27]	2004	• 3D アートワーク（Universal 3D（英語版） ファイル形式）のサポート • OpenTypeフォントの埋め込み • XFA 2.2 によるリッチテキストと属性のサポート (XFA 2.1 および 2.2 では以下のような分野向けに定義されている:動的な XFA フォーム, XFA 用の W3C XML 電子署名, Web サービス, XFA 'doc-literal' HTTP 経由のSOAP対応, SOAPによるWebサービスのための WSDL 定義, 等) • AES 暗号化 • PKCS#7 電子署名の SHA256対応,4096ビットまでの DSA 対応 • Nチャンネルの色空間サポート • ファイル添付の強化, 添付ファイルとの間の相互参照 • 電子署名による権利管理と改竄検出 [22] • Acrobat JavaScript Scripting Reference, Version 7.0[28]	7.0
1.7 (ISO 32000-1:2008[5][29])	Sixth (ISO first)[30]	2006 (ISO 2008)	• 3Dアートワークへの対応強化 • XFA 2.4によるリッチテキストと属性のサポート • 複数ファイルの添付 (portable collections) • document requirements for a PDF consumer application • 新しい文字列形式:PDFDocEncoded 文字列、ASCII文字列、byte文字列 • PKCS#7電子署名のSHA384、SHA512、RIPEMD160対応 • JavaScript for Acrobat API Reference Version 8.0 • (Adobe Acrobat Professional, Acrobat Standard, Reader にてAdobe が拡張した オブジェクト、プロパティについてのリファレンス) [31]	8
1.7 Adobe Extension Level 1	—	2008	• XFA 2.5 (Extensions Level 1) およびXFA 2.6 (Extensions Level 2) サポート[32]（XFA 2.6は以下のような分野向けに定義されている:XFA のセキュアな投稿、XFA Foreground (XFAF) 形式のサポート[33]）	8.1
1.7 Adobe Extension Level 3	Adobe Supplement to the ISO 32000, BaseVersion 1.7, ExtensionLevel 3[33]	2008	• 256-bit AES 暗号化 • PDF/A-2 向けXFA データセットの取り込み • Adobe Flashアプリ (SWF)、ビデオ（H.264 形式Flashを含む）の再生、音声その他マルチメディア埋め込みの改良。Flash playerとの双方向スクリプティングの実装により、ナビゲーターSWFファイルは Adobe Flex2 モジュールから普通のSWFとして認識されるようになった。 • XFA 2.5および 2.6[32] によるリッチテキスト対応, XFA 2.7 および 2.8対応[34]（XFA 2.7および2.8は以下のような分野向けに定義されている:Webサービスの認証ポリシー設定、WSDL/SOAP経由での投稿、タイプフェースの各国ロケール対応等）	9
1.7 Adobe Extension Level 5	Adobe Supplement to ISO 32000-1, BaseVersion:1.7 ExtensionLevel:5[35]	2009	• XFA 3.0[36]	9.1
1.7 Adobe Extension Level 6	—	2009	• XFA 3.1[37]	9.1
1.7 Adobe Extension Level 8	[38]	2011	• XFA 3.3 (Flash/SWF と XFA の統合),[39] AES-256によるパスワード暗号化方式をExtension Level 3から変更。チェックのアルゴリズムに弱点が見つかったことへの対応[40]。[41] • 仕様書は 2014年4月現在公開されていない。[29]	X (10) , XI (11)

ISO規格化 [編集]

国際標準化機構 (ISO) によるPDF 1.7の標準化以降、仕様の改訂はISOが預かるところとなった。アドビ独自のバージョンアップは、ベースとなるバージョン (1.7) に対する "Adobe Extensions" という形で行われている [42][29]。

PDFの関連規格 [編集]

ISO 15929 および ISO 15930 標準 - 2001年策定

PDFを元に策定された、印刷用途を目的としたファイル形式。印刷時のデータ交換をスムーズにするため、通常のPDFで使える機能を一部制限している。

一連のISO 19005 標準 - 2005年策定

PDFを元に策定された、電子文書の長期保存を目的としたファイル形式。主に印刷目的として利用されていたPDFを、長期保存用に特化させたもの。ISO 19005 標準は特に欧州を中心に使われており、対応するソフトウェアも欧州製のものが多い。

ISO 24517 - 2008年策定

PDFを元に策定された、エンジニアリングワークフローにおける使用を目的としたファイル形式。知的権利の安全な配布やCADデータなどの複雑な3次元データなどをPDFに組み込むことを目標にしている。

PDF/H

ISO未策定。ヘルスケアに関するデータを交換、保存するのを目的としたファイル形式。[43]

ISO 14289-1 - 2012年8月策定

ユニバーサルアクセスへの対応を目的としたファイル形式。視力や運動能力に障害のある人にも利用できるように特化させたもの。[44]

ISO 16612-2 - 2010年策定

可変データやトランザクション文書を扱うのを目的としたファイル。[45][46]

PAdES

ISO32000-2に含まれる予定。PDF文書の長期保管を目的としたPDFの拡張。欧州電気通信標準化機構により策定ならびに公開される予定。

PDF 1.7 [編集]

PDF 1.7の仕様書の最終版は 2008年1月に ISO Technical Committee 171にて採択され,2008年7月1日にISO 32000-1:2008 *Document management – Portable document format – Part 1:PDF 1.7*として公開された。

ISO 32000-1:2008 はPDFの機能を全て定義した初の規格である。それ以前にISO PDF規格になったもの (PDF/A, PDF/X, 等) は特定分野に適用される物である。ISO 32000-1はAdobe PDF仕様1.0から1.6で定められた物を網羅しているが、以前のバージョンの一部の機能はアドビにより削除されており、PDF 1.7についても同様である。[5]

ISO 32000-1の文書はアドビの PDF Reference, sixth edition, Adobe Portable Document Format version 1.7, November 2006 を元にしており、*ISO Technical Committee 171 (ISO/TC 171), Document management application, Subcommittee SC 2, Application issues*によるファストトラック (Fast-track procedure) 制度[注 2][47]により審査された。

ISO PDF 規格の要約 (abstract) には以下のように記述されている。:[48]

> ISO 32000-1:2008 は電子文書のユーザーが交換と閲覧を行うにあたり、作成した環境に依存することなく再現し、閲覧および印刷するための形式を定義する。その目的とするところはPDFファイルを作成するソフトウェア、既存のPDFファイルを閲覧およびコンテンツを画面に表示するよう解釈しユーザーと対話するソフトウェア、またその他の用途のためにPDFファイルを読み書きする各種製品の開発者に供するものである。

ISO 32000-1 の中にはアドビのプロプライエタリな仕様の部分がいくつか引用規格とされており、（Adobe Acrobat JavaScriptやXFA (XML Forms Architecture) など) ISO 32000-1[5]に準拠するためにはこれらが不可欠となっている。

PDF 2.0 [編集]

新しいバージョンのPDF規格は*ISO/CD 32000-2 - Document management – Portable document format – Part 2:PDF 2.0*[49]として策定され、2017年7月に公開された。[50]

新機能 [編集]

新機能としては以下のようなものが紹介されている。[51]

- 暗号化された文書を非暗号化された文書でラップできる。すなわち、だれにでも読める表紙ページが付いたセキュアなPDF文書を作成できる
- リッチメディア注釈の導入。地理情報やPRC(3Dフォーマット)のサポート
- PDF/A-3にて最初に導入された、添付ファイルについての機械解釈可能なメタデータが付加できるようになった
- タグ付きPDFに関する14.8章の完全な作り直し。タグセットの見直し、名前空間の導入、MathMLのサポートと発音ヒントの導入など。アクセシビリティの向上に伴いテキスト抽出や軽量デバイス向けのHTMLにコンバートするのが容易になるなど。
- デジタル署名が最新の規格に適合するようアップデートされた。ユーザーにデジタル署名されたPDFの検証が可能になった。
- 現時点でのスタンダードであるAES-256暗号化のサポート

拡張された機能 [編集]

既存の仕様から改良されたものは以下の通り。

- 注釈について透過、合成モード属性の追加
- `Polygon/Polyline real paths`
- 256bit AES暗号化
- ECC 準拠の証明書
- Unicode準拠のパスワード
- Document requirement extensions
- ページ単位、および外部から指定可能な出力インテント
- 添付ファイルのサムネイル
- 画像、フォームの XObjects に対する Measurement & Point Data
- ビュワーに対し印刷倍率の設定を強制できる
- `GoToDp` および `RichMediaExecute` アクション
- `GoTo` および `GoToR` を拡張して一部の構造化エレメントへのリンクが可能に
- 署名フィールドのロックと署名シード値の拡張
- 三次元寸法および3D表示条件の拡張（透過を含む）
- 構造化要素の追加。`DocumentFragment`、`H`*n* (深度6以上の見出し)、`Title`、 `FENote`、`SUb`、`Em`、`Strong`、`Artifact`
- 出力インテントの拡張 (`MixingHints` および `SpectralData`)

廃止された機能 [編集]

- XFA
- ムービー、サウンド、(PDF/Xの) `TrapNet` 注釈(リッチメディア注釈により代替)
- ムービー、サウンド アクション
- `Info` 辞書
- DRMについての制限
- `ProcSet` (PDFページ記述演算子)
- ファイル仕様とLaunch アクションに関するOS依存の仕様部分
- `XObjects` とフォント の名前
- 合成モードのアレイ（配列）
- 代替プレゼンテーション
- Open prepress interface (OPI)
- `CharSet` (For Type 1 fonts)
- `CIDSet` (for CID fonts)
- Prepress viewer preferences (`ViewArea`、`ViewClip`、など)
- `NeedAppearances` (フォームの見え方についての指定)
- `adbe.pkcs7.sha1` および `adbe.x509.rsa_sha1` 署名ハンドラ
- FDF ファイルの暗号化
- Suspects flag in MarkInfo dictionary

- UR signature

PDF 2.0の提案は2009年にISOに受理された。(ISO/NP 32000-2) TC 171 SC 2 WG 8 委員会は文書の作成のほか、寄せられる多数の技術的、編集面のコメントへの対応、各種課題ごとに関心を寄せる組織たちで構成される*アドホック*な下位委員会8つの運営などのために活動している。 規格の開発は"照会段階"/"国際規格案"の段階まで進んでいたが、文書の整備に本プロジェクトは中止され、新しいプロジェクトが立ち上げられた。2014年中に2つめの"委員会原案"が提出される予定である。[49][52][53]

Adobeは*Adobe Extension Level 5* と *Adobe Extension Level 3* をISO 32000-2 に盛り込もうとしたが、そのうち一部の機能だけが採用されている。

ISO委員会のPDF 2.0開発の目標には、機能の進化的な拡張のほかに、PDF言語のリファインや現在[いつ?]使われなくなっている機能（例:フォームのXObject names等）の廃止による、またアドビのプロプライエタリな仕様（例:Adobe JavaScript、リッチテキスト等）の標準化が含まれている。[54][55]

PDF 2.0ではアドビの XML Forms Architecture 3.1 を参照しようとしており、2011年にISOの委員会はアドビにXFA仕様書の提出を求め、またXFA仕様を安定化させるよう求めている[56]。

ISO TC 171 SC 2 WG 8 [編集]

（TCは「専門委員会」、SCは「分化委員会」、WGは「作業グループ」の略）

ISOの組織については国際標準化機構を参照

2008年にISO標準を策定するために設置された。

作業グループ8は年に二度会合を開き、10カ国以上より参加者が集まる。ISO 32000の委員会のミーティングは認定された主題専門家に対しオープンとされている。関心を持つ団体は、ISOの会員団体に参加について問い合わせる。

現在[いつ?]のプロジェクトのリーダーシップはシェリー・エクホルム（マイクロソフト） 、副リーダーはダフ・ジョンソン（独立コンサルタント）が担当している。 以前のリーダーは 2008年より2011までジェイムズ・キング博士（アドビシステムズ）が担当した。ISO幹事はベッツィー・ファニング(AIIM)。

技術的な基盤 [編集]

PDFの読み書きを行うソフトウェアは誰でも開発することができ、アドビにライセンス料などを支払う必要はない。アドビはPDFについて特許を保有しているが、仕様は無料で自由に参照することが出来る。[57]

PDFは3つの技術的な柱から成り立っている。

- レイアウトとグラフィックを生成するためのPostScriptページ記述言語のサブセット
- 異環境での表示のためのフォントの埋め込み、代替の仕組み
- 文書、グラフィック、添付ファイルを単一ファイルにまとめて圧縮する構造化ストレージ

PDFとPostScript [編集]

PDFは、アドビシステムズが開発し印刷業界の標準として普及していた ページ記述言語PostScriptを元に策定された。

PDFでは、コンピュータ上でのデータ交換のために次の機能が追加されている。

- ファイルに含まれる各ページへのランダムアクセス に対応。この機能により、必要なページをすばやく表示できる。PostScriptでは前のページの処理結果がないと現在ページの描画が出来ない場合があるが、PDFでは各ページで完結したコンテンツにされる。
- フォントの埋め込み
- 文書情報など、本文以外の情報を入力できる。PDFではしおり・リンク・注釈なども本文とは別の情報として扱われる
- 透明の概念（後にバージョン1.4で追加）

PDFには、PostScriptの持っているプログラミング言語としての機能はなく、HTMLなどと同様の データ記述言語となっている。たとえばPostScriptの `if` や `loop` といったプログラム用命令は取り除かれている。一方 `lineto` などは残されている。これはファイルを開いた場合にエラーが発生する可能性を小さくし信頼性を高めるためという事と同時に、PostScriptが持つ特徴の一つである、インタプリタによる実行環境への依存性を極力排除していく方向性からである。

このように元々PDFはPostScriptから発展・派生したという経緯を持つため、PostScriptとPDFは似た特性を持っており、相互の変換は比較的容易である。実際、Acrobatに含まれるDistillerでは、各種アプリケーションのデータをいったんPostScriptファイルに変換（WindowsやMacintoshではプリンタードライバを経由する形で行われる）し、それを元にPDFを生成している。しかし、PDFを作成するには、必ずしもPostScriptを経由する必要はなく、例えばGDI経由で直接PDFを作成することも可能であり、実際にそういった形で動作（GDI→PDF）する製品は多数存在する（「いきなりPDF」もこのタイプである）。

旧バージョンのQuark XPressなどからPDFファイルを作成するには、PostScriptファイルを生成したうえで"Distiller"という Adobe Acrobatに添付のソフトを使ってPDFファイルに変換するのが一般的である。また、Adobe InDesign、Illustrator、PhotoshopやQuark XPress（バージョン6以降）を使うと、Acrobatなどは使わずに直接PDFファイルを作成できる。

技術概要 [編集]

PDF文書は1以上のページで構成され、各ページにはテキスト・画像・図形が含まれる。

PDFファイルの生成には、アドビシステムズ社純正の「Adobe Acrobat」や サードパーティー製品を利用する方法がもっとも基礎的である。さらに、クラウド形態などによりサーバサイドでPDFファイルを作成するシステムや、PDFファイル生成とは異なる主目的をもつソフトウエアが PDFファイルを生成する補助機能を搭載している場合もある。

PDFファイルの表示や印刷は、第一にAcrobat Readerなどの専用のビューアーアプリケーションを用いる方法がある。ウェブ上で公開されているPDFファイルについては、パソコンの ウェブブラウザ上で閲覧できることがある。当初はAcrobat Readerのプラグインを用いていたが、現在[いつ?]Google Chrome[58]、Mozilla Firefox[59]、Microsoft Edgeなど、PDF表示機能を内蔵する製品も登場している。しかし、これらの製品でも、スマートフォン向けでは、PDF表示機能が除外されている。

PDFの仕様は、ISOで標準化される以前からアドビシステムズから公開されてきている[60]。そのため、アドビシステムズ以外のさまざまな企業・団体や有志が、PDF関連のソフトウェアを開発・公開しており、オープンソース ソフトウェア、フリーウェアも数多い。

ファイル構造 [編集]

PDFファイルはCOS ("Carousel" Object Structure) のサブセットである。COSのツリー構造は以下の8種のオブジェクトで構成される。

- ブーリアン値
- 数値
- 文字列

- 名前
- 配列:オブジェクトに順序を与えた集合
- 辞書:オブジェクトに名前を与えた集合
- ストリーム:大きなデータに使われる
- ヌルオブジェクト

オブジェクトは*object number* と*generation number*で番号付けされ、*xref table*表が各オブジェクトのファイル先頭からのオフセットを記述する。

この利点はファイル内のランダムアクセスを容易にすることで、またファイルの変更があった際に変更点を追記するだけでファイル全体の書き直しを行わなくてよくなる。 PDFのデータ配置は リニア(*最適化*)とノンリニア(*最適化なし*)の2種類がある。最適化なしは、PDFの変更箇所がファイル末尾に追記された状態であり、デメリットはファイル全体をダウンロードしてからでなければファイルの表示内容を確定できないことである。最適化された場合は、末尾にあった変更点が各ページに適用されて、Webで表示させる場合などに好きなページから表示させられるメリットがある。

イメージモデル [編集]

PDF内の画像の記述方法はPostScriptに近く、相違箇所は透明の有無である。透明はPDF 1.4で追加された。

ベクター画像 [編集]

ベクター画像は*パス*で記述された物で、直線、方形、ベジエ曲線で構成される。 図形を記述するだけでなく、文字のアウトラインを記述する場合もある。 パスはストローク、塗りつぶし、クリッピング（切り抜き）に使われる。ストロークと塗りつぶしは自由な色、模様を指定できる。 模様はタイリング、色なしのタイリング、シェーディングを指定できる。

ラスター画像 [編集]

ラスターイメージ（*Image XObjects*と呼称）は、PDF内では辞書と、関連づけられたストリームで表現される。辞書には画像のプロパティが記述され、ストリームにはイメージデータが記述される。画像は複数の圧縮方式で *フィルター*される。サポートされるフィルターは以下のものがあり、用途に応じて画質とサイズのバランスを選ぶことができる。

`ASCII85Decode`
: ストリームをASCIIの7ビットで表現するもの

`ASCIIHexDecode`
: ASCII85Decodeに近いがよりコンパクトになる

`FlateDecode`
: zlib/deflate（いわゆるgzip形式。ZIPではないので注意）アルゴリズムで圧縮するもの。;PDF 1.2より実装

`LZWDecode`
: LZWアルゴリズムで圧縮するもの。

`RunLengthDecode`
: ランレングスアルゴリズムによるシンプルな圧縮

`DCTDecode`
: JPEGに採用された 非可逆圧縮

`CCITTFaxDecode`
: CCITT (ITU-T) faxで規定された 可逆 2値 (黒/白)圧縮。Group3とGroup 4がある

`JBIG2Decode`
: JBIG2標準で定義された黒白二値の形式。可逆、非可逆が選べる。PDF 1.4より実装

`JPXDecode`
: JPEG 2000標準で定義された形式。可逆、非可逆が選べる。PDF 1.5より実装

テキスト [編集]

テキストはPDF内のコンテント ストリーム内で *テキスト エレメント*として記述される。テキストエレメントはどの文字がどの座標に表示されるかを定義し、どのフォントとエンコーディングを使うかも記述する。

フォント [編集]

フォントオブジェクトは、タイプフェースを記述する。どのフォントを使用するか記述するほかに、フォントファイルの埋め込みもできる。 埋め込めるフォントはType1、TrueType、OpenTypeがある。

標準Type1フォント（標準14書体） [編集]

以下の14書体については、ソフトウェアは必ず装備するか、同じ寸法の代替フォントを装備しなければならない。ソフトウェアは以下のフォントの寸法について情報を保持し、もしPDFに埋め込まれていない場合は同じ寸法の代替フォントで表示する。

- Times (v3) (標準、斜体、太字、太字斜体)
- Courier (標準、斜体、太字斜体)
- Helvetica (v3) (標準、斜体、太字斜体)
- Symbol
- Zapf Dingbats

フォントの埋め込みの注意 [編集]

電子文書を正しく表示するためには、フォントが正しく設定されている必要がある。一般に、文書作成時に使用されているフォントがインストールされていないコンピュータでは、文書を正しく表示・印刷できない。例えばヒラギノフォントを使って作成した文書

は、このフォントがインストールされていないコンピュータでは代替の日本語フォントで表示する必要がある。さらに、日本語フォントがインストールされていないコンピュータではエラーや文字化けが発生し、正しく表示できない。

PDFの文書では、使用しているフォントを埋め込むことで、そのフォントがインストールされていないコンピュータでも正しく表示・印刷できる。フォントを埋め込む方法は2つあり、当該フォントに含まれているすべてのグリフ（字形）を埋め込む方法と、文章に使用されているグリフのみを埋め込む方法である。これらの選択は、PDFを作成する際に行う。フォントを埋め込んで作成したPDFの日本語文書は、日本語フォントがインストールされていないコンピュータでも正しく表示できる。

ただし、フォントを埋め込んだPDFファイルはファイルサイズが大きくなるという問題がある。また、フォントを埋め込む場合は、フォントのライセンスにも注意する必要がある。

エンコーディング [編集]

テキストストリーム内では、文字は文字コード（番号）によって記述される。WinAnsi、MacRomanのほか東アジアの各種言語のエンコーディングが定義されているほかに、フォント自身に独自のエンコーディングを持つことも出来る。 基礎がType1フォントを前提に作られたので、TrueTypeフォントなどを記述する場合は、非常に複雑な記述になる。

日本語のエンコード [編集]

文字コードについては、全ての文字列が独自のルールで英数字にエンコードされる。英語文書の場合はある程度解読できるが、多バイト文字になると非常に複雑なルールでエンコードしなければならない。

それで得られるメリットは、処理系にかかわらず文字を正確に区別できることで、例えばJIS90形式の文字とJIS2004形式の文字はJIS、Unicode上では同じ文字コードでありながら字形が異なる文字があり、普通の処理系ではどちらかのフォントしか表示できないが、PDF上では一緒に記述できることが挙げられる。[61]

しかしながら例えばWindows 2000とWindows Vista間ではMS明朝、MSゴシックのフォントのバージョンは異なり、フォントを埋め込まないようにPDFを作成すると、片方の環境では文字が全面的に文字化けするケースがある。

また、フォントが埋め込まれているため文字化けしていないように見えても、文面をクリップボードにコピーしてエディタに貼り付けると、テキストストリームが異機種のエンコーディングになっていて文字化けするケースもある。

Adobe製品は比較的問題が起こりにくいが、サードパーティー製品や海外製のソフトウェアでは現在[いつ?]もこの種の問題に遭遇することがある。

インタラクティブ性 [編集]

PDFファイルには、音楽、動画などのマルチメディアファイルを含めることができる。そのためPDFファイルは、コンピュータを使ったプレゼンテーション用に使うこともできる。2005年にアドビシステムズがFlashの開発・推進を進めてきたマクロメディア を買収し、アドビシステムズによるFlashとPDFとの統合がかつて進められていた。Flashは2020年に開発・サポートが終了している[62]。

そのためAdobe製品によるPDF表示では、Flash形式のファイルが埋め込まれていてもプラグインなどをインストールせずに再生することが出来る。しかしQuickTimeはサポートされない。これはAppleとAdobeの敵対的な関係を示しており[要出典][63]、そのためAppleのiOSデバイスではリッチメディアPDFはインタラクティブ性が損なわれる。

またフォームはユーザーに入力させるメカニズムで、AcroForms(PDF 1.2より)と Adobe XML Forms Architecture(XFA)(PDF 1.5より) の2種類が使える。[64]XFA仕様はPDF仕様に含まれておらず、オプション機能として言及されているだけである。

AcroForms [編集]

PDF 1.2より導入され、テキストボックスやラジオボタンなどのオブジェクトとJavaScriptを記述することが出来る。PDFの標準のアクションに加え、AcroForms は内容の送信、リセット、取り込みができる。送信機能はフォームの内容を特定のURLに送信する。 送信形式は以下の種類がある。

- HTML Form format（HTML 4.01 形式は PDF 1.5より;HTML 2.0 形式は PDF1.2より）
- Forms Data Format (FDF)
- XML Forms Data Format (XFDF)（external XML Forms Data Format Specification, Version 2.0;PDF 1.5より;PDF 1.4で実装された"XML" フォーム送信形式の代替）
- PDF （PDFの内容全部を送信できる）（PDF 1.4 にて定義）

AcroFormsはフォームフィールドの値を外部の独立したファイルに キー:値 のペアで格納することができる。外部ファイルとしてはForms Data Format (FDF) とXML Forms Data Format (XFDF) の2種類がある。 [65][66][67]

Forms Data Format (FDF) [編集]

Forms Data Format (FDF)	
拡張子	.fdf
MIMEタイプ	application/vnd.fdf[68]
タイプコード	'FDF'
開発者	アドビ
初版	1996年 (PDF 1.2)
派生元	PDF
拡張	XFDF
国際標準	ISO 32000-1:2008
オープンフォーマット	Yes

Forms Data Format (FDF) はPDFを基盤にしており、同じ文法と基本的に類似したファイル構造をしているがPDFよりはbodyに必要なオブジェクトが1個だけである分単純である。FDFはPDF仕様書に記載されている。(PDF 1.2より) FDF形式はデータをサーバに送り、応答を受け取り、フォームに内容を反映するのに使用できる。また単体ファイルとしてエクスポートさせ、PDFのインタラクティブ フォームに取り込む使い方も出来る。 PDF 1.3からは、FDFはアノテーション（注釈）を適用対象のPDFから独立して格納するコンテナーとして定義できるようになった。典型的なFDFはX.509 certificatesや認証リクエスト、ディレクトリ設定、タイムスタンプサーバー設定、その他添付のPDFファイルなどの情報をネットワーク送信のためカプセル化する。 [67] FDFのMIMEコンテントタイプは application/vnd.fdf で, ファイル拡張子は.fdfになる。Macのファイルタイプは'FDF'。[64] FDFの単体ファイルへのインポート、エクスポートはフリーウェア系PDFソフトウェアでは広く実装しているとは言えない。その例としては、Evince、Okular、Poppler、KPDF, Sumatra PDF が挙げられる。ただしEvince、Okular、PopplerはAcroformsにデータを代入して保存させることなら出来る。単体FDFのインポートを実装しているソフトウェアとしてはAcrobat Readerがあり、インポート、エクスポート両方を実装するものとしてはFoxit ReaderとPDF-XChange Viewer Freeが挙げられる。PDFファイルへのFDFデータの保存についてはpdftkにも実装されている。

XML Forms Data Format (XFDF) [編集]

Adobe XML Forms Architecture (XFA) [編集]

論理構造とアクセシビリティ [編集]

「タグ」機能により、PDFのコンテンツにはセマンティックな（意味に基づく）構造と関連が記述できるようになり、これによりPDFの順序付けた読み上げが可能になった。

タグ付きPDFは印刷用途のPDFには必要ないためオプション扱いとなっており、ISO32000-1の記述も曖昧な物になっている。

技術の特筆点 [編集]

この節には**複数の問題があります**。改善やノートページでの議論にご協力ください。
- **出典**がまったく示されていないか不十分です。内容に関する文献や情報源が必要です。（2022年6月）
- **独自研究**が含まれているおそれがあります。（2022年6月）

スキャンした文書 [編集]

紙資料をイメージスキャナ などを使って電子文書にする場合も、PDFが利用されることが多い。

紙資料を電子文書化するとき、PDFを利用しない場合には、TIFFなどの画像ファイルとして保存する方法と、 OCRソフトウェアを使ってテキストとして保存する方法がある。画像ファイルとして保存された電子文書は画面上で見る場合には紙資料と同じ内容が再現できるものの、文章や文字をコピーすることができないなどテキストの再利用に大きな制限がある。また、OCRソフトウェアを使って作成したテキストファイルではテキストの再利用は可能となるが、OCRソフトウェアの精度の問題もあり、元の内容を完全に再現できない場合が多い。

PDFを利用すれば、紙資料をスキャンした画像の上にOCRソフトウェアで変換して作成した透明テキストを重ね、1つのファイルとして保存することができる。こうしたPDFの文書では、画面上で見る場合には紙資料の内容を完全に再現でき、不完全ではあるがテキストの再利用もできる。例えば、Acrobatにはバージョン6.0以降のStandard版以上でOCR機能が標準で搭載されるようになった。

アクセシビリティ [編集]

PDFの役割は印刷物をスクリーンに再現するだけではない。視覚障碍者向けの対応として、バーチャルな印刷物にとどまらず、XMLによる"タグ"を埋め込むことが出来る。これはページ内コンテンツの論理的な構造を表し、読み上げソフトを使う場合にどの順序で読むべきかなどの情報を提供する。PDF作成用ソフトウェアによってタグ生成への対応度は分かれ、またデフォルトでタグ生成を有効にしていないソフトもある。特に紙からスキャンしてPDFを生成する場合はタグは軽視され、障碍者にとって障壁となることが多い。

PDFのアクセシビリティ対策は(i)フィジカルなビュー（普段目にする印刷物的な表示）と、読み上げソフトウェアが使用する(ii)タグのビュー、そして(iii)PDFを構成するテキストやグラフィックのオブジェクトのストリーム（タグがない場合には読み上げソフトはこのテキストストリームを読みにいく）という3つのビューの整合性を取るという困難な作業を伴う。

ウイルスと脆弱性 [編集]

閲覧に使用したコンピュータのウイルス感染 [編集]

PDFはセキュリティが考慮されていない古い時代のフォーマットである。脆弱性のあるAcrobat Readerで有害なJavaScriptを含むPDFファイルを開くと、コードが実行され、パソコンがウイルスに感染する。同時に、ダミーのPDFファイルを表示させて、ユーザーがウイルス感染を気付かないようにする。その脆弱性を悪用した攻撃があり、2009年12月以降、日本で話題になっている「Gumblar」もPDFを媒介としている。

対策として、リーダの最新版へのプログラムアップデートがあるが、新たな脆弱性が発見されてからアップデートまでに時間が必要である。ユーザーによる不要な機能の無効化（例えばAcrobat ReaderにおけるJavaScriptエンジンの無効化）を併用することが推奨される。

権利コントロール [編集]

セキュリティと電子署名 [編集]

PDFファイルには、情報の機密性を保つために、閲覧パスワード（ユーザパスワード）と編集パスワード（オーナーパスワード）、公開鍵方式の暗号化を設定することができる。

閲覧パスワードが設定されていると、利用者は正しい閲覧パスワードを入力しないとPDFファイルを開けない。編集パスワードが設定されている場合、閲覧だけならパスワード入力は不要であるが、次の作業をするには正しい編集パスワードを入力して設定を解除しなければならない。

- 編集
- 印刷
- テキストや画像などのコピー

この機能を使うことにより、ユーザの画面上では表示できるものの、コンテンツ内の文章をコピー・アンド・ペースト できないようしたり、文書内の写真の印刷ができないよう設定した文書を配布したりできる。

また、電子署名を付け、文書の改竄を防止する機能も持つ。

しかしPDFのパスワードを解除するソフトウェアも多く流通しており、2014年現在ではPDF 1.6などの古い形式のファイルは1秒もかからずパスワードが解除されてしまうケースがある。数年後にはツールで解読されてしまうということを考慮する必要がある。

DRM [編集]

Adobe LiveCycle（英語版）製品群ではDRMを実装しており、エンドユーザーに配布されたPDFに対して、後から閲覧権限を変更することができる。応用の一例としては、配布されたPDFの内容が古くなってしまった場合に閲覧禁止として、新しい版をダウンロードするよう求めるような例がある。

デフォルトの画面設定 [編集]

PDFは表示するソフトウェアの挙動を定義することが出来る。 例えばしおりのウインドウを最初に開くか、またツールバーを表示するかどうか、などを文書に記述できる。

ソフトウェア [編集]

PDFの生成、閲覧用ソフトウェアは各種プラットフォームで提供される。 macOSとほとんどのLinuxディストリビューション、LibreOffice、Microsoft Office 2007（要SP2アップデート）、WordPerfect 9、Scribus、Windows向け各種プリントドライバ、pdfTeX組版システム、DocBook PDFツール、Ghostscript製品群、アドビのAdobe AcrobatとAdobe FrameMaker、Googleのオンラインオフィススイートである Google DocsはPDFに出力する機能を装備している。

ラスターイメージプロセッサ (RIP) は、PDFをラスターイメージラスターイメージ に変換して、紙などの媒体に出力するために使用され、DTP分野ではラスタライズと呼ばれる。 PDFの処理能力を備えるRIP は、アドビのPDF印刷エンジン[69] または、サードパーティー製品 Jaws[70] や Global Graphics社（英語版）のHarlequin（ハーレクイン）RIP[71] を組み込んでいる。

PDFファイルの閲覧と印刷 [編集]

Windows環境におけるPDFファイルの表示や印刷には、アドビシステムズから無料で配布されているAcrobat Readerを使うのが一般的である。Acrobatがインストールされている場合は、AcrobatでPDFの表示や印刷ができる。macOSではOSに標準で付属

する「プレビュー」を利用できる。その他のOSについても、PDF閲覧ソフトに付属している印刷機能や、OSの印刷機能を利用して印刷できる場合が多い。

PDFファイルの検索 [編集]

Web上のPDFファイルは、Googleなどで検索できる。また、コンピュータ内のPDFファイルは、AcrobatとAcrobat Readerによる全文検索が可能だが、検索用インデックスを作成した高速全文検索を利用するためにはAcrobatのProfessionalバージョン（6.0以降）やGoogleデスクトップ、Mac OS X Tiger以降に付属するSpotlightなどが必要となる。

PDFファイルの作成と編集 [編集]

PDFファイルの作成には、アドビシステムズのAcrobatを利用するのが一般的である。macOSではOSの標準機能で各種文書をPDFファイルに変換できる。LinuxなどUnix系OSの印刷システムであるCUPSにはPDFファイルの出力機能がある。そのほかにも、後述するLibreOfficeなどオープンソース のものも含めて、数多くのPDF作成ツールがある[72][73]。

Acrobat [編集]

Acrobatでは、データを各種ソフトウェアから「Adobe PDFプリンタ」へ印刷することでPDFファイルを作成できる。この操作の場合、Acrobatに含まれるDistillerでPDFファイルを作成することになる。また、Microsoft OfficeではAcrobatに含まれるPDFMakerで文書をPDFに変換できる。PDFMakerはDistillerを呼び出すとともに、しおり・ハイパーリンク・注釈などを自動的に作成する。

Adobe PDFプリンタによる方法以外としては以下のような作成手法を備えている。

- Acrobat から直接、単数もしくは複数の画像ファイルを指定して、PDF化することが出来る。市販のデジタル写真集などでも利用されている。
- Web Capture機能によりウェブページを直接PDF変換する。階層を指定することでハイパーリンク構造も再現できる。
- イメージスキャナ から直接画像を読み取り、PDFに変換できる。

様々な文書データのPDF化 [編集]

Microsoft Officeや一太郎などで作成した文書データも、PDF化されることが多い。PDFの文書は、Microsoft Officeなど文書作成時に使ったソフトウェアをインストールしていないコンピュータでも表示・印刷でき、コンピュータの環境によってレイアウトが変わる可能性も小さくなる。

Microsoft Officeの文書はPDF化しなくても、マイクロソフトから無償配布されている表示専用ソフトウェア（Word ViewerやExcel Viewerなど）で表示させることができる。しかしこうしたソフトウェアは、Acrobat ReaderなどのPDF表示用ソフトウェアと比べると、対応しているOSが限られていることもあり、インストールされていない、またはできない場合が多い。そのため、不特定多数の人を対象にした文書はPDF化することで正しく表示される可能性が高くなる。

PDFの作成には、Microsoft Officeからはプリンタとしてインストールされる「Adobe PDF」や「Acrobat Distiller」を利用してPDFを作成することができる。なお、32bit版Officeについては、Acrobatに含まれるマクロの「PDF Maker」を利用し、より簡単なPDFの作成が可能になっている。

2007 Microsoft Office System (Microsoft Office 2007) では追加アドインを加えることでPDFを出力する機能が追加された。また、Microsoft Office 2007 サービスパック 2では標準機能として追加され、別アプリを利用することなしにPDFを作成することができる。このPDF作成機能はMicrosoft Office 2010 (x86 / x64) にも引き継がれている。

一太郎の「2011 創」以降のバージョンでは、一太郎文書から直接PDFを作成することもできる。それより前のバージョンでは、CubePDF等の無料仮想プリンタドライバを用いて作成する事ができる。

オープンソースソフトウェア のLibreOfficeは標準でPDF出力に対応しており、PDFにODFファイルを埋め込みLibreOfficeで編集可能なPDFを作成するハイブリッドPDF[74]や長期保存向けのPDF/A、ユニバーサルアクセシビリティ対応の PDF/UA[75]での出力や、PDFへの電子署名にも対応している。また、LibreOffice Drawを利用して、PDFファイルを直接、編集したり機密情報を消すための墨消しも行える。

LaTeXとPDF [編集]

LaTeXで作成した文書をPDFに変換する機能も持つツールも開発されている。

- PDFLaTeXはLaTeXソース文書を読み取り、そのままPDF形式に出力できる[注 3]。
- dvipdfm(x)はLaTeX標準の中間形式であるdvi形式のファイルをPDFに変換できる。
- これらPDFを直接扱う方法ではなく、LaTeX標準のdvipsやdvipskなどの伝統的なPostScript出力用ツールでいったんPostScript形式に落とし、それをAdobe Distiller（または自由ソフトウェアであるGhostscript）といった標準的なPostScript→PDF変換ツールを使ってPDF出力させるといったやり方も一般的である
- ProsperなどのLaTeX形式ファイルからプレゼンテーション用PDFファイルを生成できるツールも存在する。

XML文書のPDF化 [編集]

マークアップ言語XMLの応用技術であるXSL-FOを利用すると、Apache FOPやXSL Formatterなどのソフトウェアを利用してPDFファイルに変換できる。XSL-FOはXSLTなどを利用して各種XML文書から生成できるため、XSL-FOを利用することで各種XML文書からPDFファイルを作成できる。

PDFの短所 [編集]

> この節には**複数の問題があります**。改善やノートページでの議論にご協力ください。
> - **出典**がまったく示されていないか不十分です。内容に関する文献や情報源が必要です。（2013年3月）
> - ほとんどまたは完全に**一つの出典**に頼っています。（2019年5月）
> - **独自研究**が含まれているおそれがあります。（2019年5月）

PDFの短所として、次のような点が指摘されている。

仕様が複雑 [編集]

PDFは、元にしていたPostScriptの仕様が複雑だったため、簡略化したとはいえやはり複雑な仕様である。また、PDFのバージョンアップとともにさまざまな機能が追加されたため、仕様はますます複雑になっている。

このため、PDFの文書およびソフトウェアはサイズが肥大化する傾向にあり、ソフトウェアの動作が重く、メモリの使用量が増大しがちである。

例えば、Acrobat6ではソフトウェアのサイズが大きく、起動に時間がかかるなど動作が重いため、敬遠する人も多かった。Acrobat7になって起動時間は短縮したが、ソフトウェアのサイズは相変わらず大きく、動作の重さを感じる場面もある。その配慮として、PDF以外の形式による文書（Word文書、Excelワークシートなど）も公開しているサイトもある。

また、仕様が公開されているとはいえ、PDFの複雑な仕様に完全に対応するソフトウェアを作成するのは難しい。

再利用が困難 [編集]

PDFは、文字のレイアウトが崩れにくい。作成社の意図がこれ以上改変されていないという保証を担保した最終成果物として、限定または不特定多数の利用者に公開するのには適している。しかし文字列の抽出や他形式にコンバートを試みると困難に直面する。というのは何も加工されていないPDFでは、ある文字列のブロック、隣接または上に重なる文字列やグラフィックとの論理的な対応関係を示す情報が乏しいためである。 言い換えれば、PDF上では文字列が、変換元文書にあったコンテキストを無視した文字の羅列になってしまうのである。これはKindleをはじめとする、画面の小さいビューアー用フォーマットへの変換を困難にする。また視覚障碍者向けの拡大表示、読み上げなどアクセシビリティにとって障壁となる。

この問題点に対応するため、PDFには「タグ」機能が「Adobe Acrobat 2017」で追加され、PDFへの変換時に文字列がどの順番で読まれるべきか、見出しとの関係などの情報が追加されるように改善された。しかし、紙をスキャナー機でスキャンまたはスマホ上で撮影して作成されたPDFではタグは軽視されることが多い。

HTMLとの比較 [編集]

PDFは画面で見るには、ユーザビリティが不十分な側面がある。特に、スマートフォン普及以前の規格であるため小型端末では扱いにくい[76]。ユーザビリティ（閲覧しやすさ）に関して、ヤコブ・ニールセンはPDFを「オンラインの閲覧用に使ってはならない」、と結論づけている[77]。

- ユーザビリティに十分配慮して作成されたHTML文書と比べると、PDFは扱いにくい面がある。PDFはウェブブラウザーでの表示用に最適化（リニアライズ）されていないと、文書の一部分だけを参照したい場合でも、最初から最後まですべてのデータを閲覧端末に読み込む必要がある。Acrobatなど既定でウェブブラウザーでの表示に最適化したPDFを作成するソフトも多いが、廉価・無償のPDF作成ソフトではウェブブラウザーでの表示に最適化する機能がないものがある。このようなPDF作成ソフトで作成されたPDFをウェブブラウザーで表示すると、表示開始までの待ち時間が長くなる。
- ナビゲーションのために、しおり、PDFのページ間（内部）リンクやPDF外部へのリンクを文書の任意の箇所に設定することも可能であるが、これを利用するには、PDF作成時に素材データの中で設定するか、（Readerでない）Acrobatなどのしおり・リンク編集機能をもつソフトで追加する必要がある。この点は、ソースに参照したい箇所をテキスト情報として付記するだけで済むHTMLに比べれば煩雑である。
- ほとんどのPDF文書がA4縦長で作られているが、PC画面は横長であることが多く、A4縦長文書を等倍で表示させにくい。ディスプレイは印刷物に比べて解像度が低いため、寸法の小さいパーツは見えにくい。HTMLは、ブラウザがウィンドウのサイズに合わせて再整形するのでこのような問題は少ない。
- PDFの長所であったフォント埋め込みに関しても、Webフォントが登場したためネット通信があればHTMLではWebフォントを使用可能。
- アクセシビリティの観点からも、PDFでは文書の作成時にタグ付きPDFとしなければならない点などを考えると、HTMLや単純なテキスト形式の方が扱いやすい。

脚注 [編集]

[脚注の使い方]

注釈 [編集]

1. ^ ただし、それ以前からフリーソフトウェアによる（日本語を含む）PDFの作成は可能であった。
2. ^ ISO/IECのファストトラックは、既に標準化された既存の規格に基づき、通常よりも迅速に審議を行い国際標準化する方式。
3. ^ 日本語を含んだLaTeXソース文書は扱えない。

出典 [編集]

1. ^ “RFC 8118 - The application/pdf Media Type” (英語). 2017年3月20日閲覧。
2. ^ “ISO 32000-1:2008 Document management — Portable document format — Part 1: PDF 1.7”. 2021年3月22日閲覧。
3. ^ Yoichi Yamashita（マイコミジャーナル） (2008年7月3日). “PDF 1.7がISO 32000-1として国際標準化”. *毎日コミュニケーションズ. オリジナル*の2009年5月31日時点におけるアーカイブ。 2008年7月17日閲覧。
4. ^ 垣内郁栄（AtmarkIT） (2008年10月28日). “PDFのISO規格化で得たものと失ったもの：担当者が振り返るPDF1.7標準化の道のり”. *ITMedia* 2014年5月19日閲覧。
5. ^ *a b c d e* “ISO 32000-1:2008 - Document management – Portable document format – Part 1:PDF 1.7”. Iso.org (2008年7月1日). 2010年2月21日閲覧。
6. ^ “Macで書類をPDFとして保存する”. *Apple Support*. 2023年8月23日閲覧。
7. ^ “iPhone や iPad のブック App で PDF を保存および編集する”. *Apple Support* (2022年10月24日). 2023年8月23日閲覧。
8. ^ 吉田印刷所. “PDF形式の7つのメリット～なぜ印刷にはPDFが適しているのでしょうか？”. 特売プレス+吉田印刷所 DTPサポート情報Blog. 2017年3月13日閲覧。
9. ^ “The Camelot Project” (PDF) (英語). 2009年10月27日閲覧。
10. ^ “A CDE Definition - Common Ground” (英語). The Computer Desktop Encyclopedia. 2015年7月1日閲覧。
11. ^ “A CDE Definition - Farallon Replica” (英語). The Computer Desktop Encyclopedia. 2015年7月1日閲覧。
12. ^ Laurens Leurs. “The history of PDF” (英語). 2007年9月19日閲覧。
13. ^ Geschke, Charles, *Driving Adobe:Co-founder Charles Geschke on Challenges, Change and Values*, The Wharton School of the University of Pennsylvania
40. ^ *PDFlib - PDF Security - Encryption Algorithms and Key Length* 2012年9月26日閲覧。
41. ^ *PDFlib - PDF Security - Security Recommendations* 2012年9月26日閲覧, "AES-256 according to PDF 1.7 Adobe Extension Level 3 (Acrobat 9) should be avoided because it contains a weakness in the password checking algorithm which facilitates brute-force attacks against the password. For this reason Acrobat X no longer offers Acrobat 9 encryption for protecting new documents (only for decrypting existing documents). In summary, AES-256 according to PDF 1.7 Adobe Extension Level 8/PDF 2.0 or AES-128 according to PDF 1.6/1.7 should be used, depending on whether or not Acrobat X is available. Passwords should be longer than 6 characters and should contain non-alphabetic characters."
42. ^ *Adobe Supplement to ISO 32000-1, BaseVersion:1.7 ExtensionLevel:5*, Adobe Systems Incorporated, (2009-06), p. 5
43. ^ [1] Implementation Guide for the Portable Document Format Healthcare
44. ^ [2] ISO 14289 の販売ページ
45. ^ バリアブルデータ印刷(Global Graphics Software社の解説)
46. ^ “ISO 16612-2:2010 - Graphic technology -- Variable data exchange -- Part 2:Using PDF/X-4 and PDF/X-5 (PDF/VT-1 and PDF/VT-2)”. 2014年9月22日閲覧。
47. ^ “ISO規格の制定手順”. 2014年5月9日閲覧。
48. ^ ISO 32000-1:2008, Page 1, section "1 Scope"
49. ^ *a b* “ISO/CD 32000-2 - Document management -- Portable document format -- Part 2:PDF 2.0” (2013年7月26日). 2017年9月1日閲覧。
50. ^ PDF Association (2017年8月3日). “ISO 32000, the PDF specification, updated to version 2.0”. 2017年9月1日閲

14. ^ Duff Johnson. "The 8 most popular document formats on the web" (英語). 2014年3月2日閲覧。
15. ^ "Adobe Developer Connection:PDF Reference and Adobe Extensions to the PDF Specification" (英語). Adobe Systems. 2017年3月13日閲覧。
16. ^ "PDF compatibility levels". 2011年10月1日時点のオリジナルよりアーカイブ。2015年6月17日閲覧。
17. ^ R, Leonard, *History of PDF Openness*, Acrobat users, オリジナルの2007-10-14時点におけるアーカイブ。
18. ^ *Portable Document Format Reference Manual*, Adobe Systems Incorporated, (1993-06) 2015年6月17日閲覧。
19. ^ a b (PDF) *Portable Document Format Reference Manual Version 1.2*, Adobe Systems Incorporated, (1996-11-12), オリジナルの2005-11-03時点におけるアーカイブ。 2015年6月17日閲覧。
20. ^ a b (PDF) *PDF Reference Second edition – Adobe Portable Document Format Version 1.3*, Adobe Systems Incorporated, (2000) 2015年6月17日閲覧。
21. ^ PDFリファレンス第2版 ISBN 4-89471-338-1
22. ^ a b c d "Adobe PDF Reference Archives". Adobe Systems Incorporated. 2010年2月23日閲覧。
23. ^ a b (PDF) *PDF Reference Third edition – Adobe Portable Document Format Version 1.4*, Adobe Systems Incorporated, (2001) 2015年6月17日閲覧。
24. ^ *Technical Note#5186 Acrobat JavaScript Object Specification Version 5.1*, (2003)
25. ^ (PDF) *PDF Reference fourth edition – Adobe Portable Document Format Version 1.5*, Adobe Systems Incorporated, (2003) 2010年2月23日閲覧。
26. ^ *Acrobat JavaScript Scripting Guide, Technical Note#5430, Version:Acrobat 6.0*, (May 2003)
27. ^ (PDF) *PDF Reference fifth edition – Adobe Portable Document Format Version 1.6*, Adobe Systems Incorporated, (2004) 2015年6月17日閲覧。
28. ^ *Acrobat JavaScript Scripting Reference*, (2005-06-27)
29. ^ a b c "Adobe Developer Connection:PDF Reference and Adobe Extensions to the PDF Specification". Adobe Systems. 2010年12月13日閲覧。
30. ^ (PDF) *PDF Reference sixth edition – Adobe Portable Document Format Version 1.6*, Adobe Systems Incorporated, (2006) 2015年6月17日閲覧。
31. ^ *JavaScript for Acrobat API Reference, Version 8*, (April 2007)
32. ^ a b *XML Forms Architecture (XFA) Specification Version 2.6*, Adobe Systems Incorporated, (2008-01-25) 2014年4月9日閲覧。
33. ^ a b *Adobe Supplement to the ISO 32000 BaseVersion:1.7 ExtensionLevel:3*, Adobe Systems Incorporated, (2008-06) 2014年4月9日閲覧。
34. ^ *XML Forms Architecture (XFA) Specification Version 2.8*, Adobe Systems Incorporated, (2008-10-23) 2014年4月9日閲覧。
35. ^ *Adobe Supplement to the ISO 32000 BaseVersion:1.7 ExtensionLevel:5*, (2008-06) 2015年6月17日閲覧。
36. ^ *XML Forms Architecture (XFA) Specification Version 3.0*, Adobe Systems Incorporated, (2009-03-12) 2014年4月9日閲覧。
37. ^ *XML Forms Architecture (XFA) Specification Version 3.1*, Adobe Systems Incorporated, (2009-11-16) 2014年4月9日閲覧。
38. ^ (PDF) *PDFlib API Reference 8.0.2*, 2011年3月7日閲覧, "1.7ext8 – PDF 1.7 extension level 8 requires Acrobat X"
39. ^ *XML Forms Architecture (XFA) Specification Version 3.3*, Adobe Systems Incorporated, (2012-01-09) 2014年4月9日閲覧。
51. ^ Duff Johnson (PDF Association). "PDF 2.0:The worldwide standard for electronic documents has evolved". 2017年9月1日閲覧。
52. ^ "ISO/NP 32000-2 - Document management -- Portable document format -- Part 2:PDF 2.0" (2009年10月6日). 2010年2月24日閲覧。
53. ^ Duff Johnson. "ISO 32000-2 (under development)". 2014年3月2日閲覧。
54. ^ Leonard Rosenthol, Adobe Systems (2012年). "PDFと各種標準(英語)". 2013年10月20日閲覧。
55. ^ Dr. Matthew Hardy, Adobe Systems (2012年). "7分でわかるPDF標準(英語) – PDF (ISO 32000)". 2013年10月20日閲覧。
56. ^ *ISO/TC 171/SC 2/WG 8 N 603 - Meeting Report*, (2011-06-27), "XFA is not to be ISO standard just yet. ... The Committee urges Adobe Systems to submit the XFA Specification, XML Forms Architecture (XFA), to ISO for standardization ... The Committee is concerned about the stability of the XFA specification ... Part 2 will reference XFA 3.1（pdf.editme.com 会員限定）"
57. ^ partners.adobe.com - Developer Resources
58. ^ 柳 英俊 (2010年12月6日). "サンドボックスで保護されたPDF閲覧機能を備える「Google Chrome」v8が正式公開". *窓の杜*. 2017年3月20日閲覧。
59. ^ Lardinois, Frederic; iwatani (2013年2月20日). "Firefox 19がHTML5だけによるPDFビューワを搭載, Androidバージョンはテーマをサポート". *TechCrunch Japan*. 2017年3月20日閲覧。
60. ^ Adobe PDF Specifications PDF仕様書の一覧ページ（英語）
61. ^ [3]
62. ^ "Adobe Flash Playerサポート終了". *www.adobe.com*. 2023年8月23日閲覧。
63. ^ John Naughton (2010年4月18日). "AppleはiPadでAdobeとFlashに宣戦（英語記事）". TechGuardian. 2014年5月9日閲覧。AppleがFlashを不安定でSafariブラウザがクラッシュする主要要因としてiPhoneから排除したこと、さらにAdobeがFlashをiOS用にコンバートするツールをCS5で提供したものに対しAppleが4日後に「中間変換/互換レイヤーの禁止」条項で応じた経緯が紹介されている。
64. ^ a b Adobe Systems Incorporated (2008-07-01), *Document Management – Portable Document Format – Part 1:PDF 1.7, First Edition*, 2010年2月19日閲覧。
65. ^ Adobe Systems Incorporated (2007年10月15日). "Using Acrobat forms and form data on the web". 2010年2月19日閲覧。
66. ^ (PDF) *XML Forms Data Format Specification, version 2*, (September 2007) 2010年2月19日閲覧。
67. ^ a b (PDF) *FDF Data Exchange Specification*, (2007-02-08) 2010年2月19日閲覧。
68. ^ *IANA Application Media Types - vnd.fdf* 2010年2月22日閲覧。
69. ^ Adobe PDF Print Engine 2
70. ^ www.globalgraphics.com/products/jaws_rip/
71. ^ Harlequin RIP
72. ^ TeX Wiki PDFの作り方
73. ^ さまざまなPDFの作成技術の概観 アンテナハウス PDF資料室
74. ^ "LibreOffice 7.1 Help PDF Export General". 2021年3月22日閲覧。
75. ^ *LibreOffice 7.0: Release Notes* 2021年3月22日閲覧。
76. ^ なぜいまだにわざわざ文書をPDFに変換して公開するのか？分からない | TechWave（テックウェーブ）
77. ^ PDF：人間が消費するには不向き（Jakob Nielsen博士のAlertbox）

関連項目 [編集]

PDFソフトウェアの一覧 [編集]

詳細は「PDFソフトウェアの一覧」を参照

PDFの競合規格 [編集]

CAJ (caj・kdh・nh・teb)

China Academic Journals Document。中国知网（CNKI（英語版））で用いられている形式。CAJViewer（CAJ全文浏览器, 中国期刊网的专用全文格式浏览器）で閲覧できる。MIME 形式は「application/octet-stream」。

DjVu（デジャヴ）

米国AT&T研究所が開発した、PDF同様に文書公開用に使われるファイルフォーマット。特に画像データの圧縮率が高く、PDFやJPGよりもサイズが小さい割に画像の劣化が少ない特徴を持つとされ、2005年頃から一部で使用されはじめている。

DocuWorks（XDW・XBD）

富士フイルムビジネスイノベーション 製の、電子文書と紙文書を一元管理するオフィス向け文書管理アプリケーション。DocuWorks（6.0以降）の文書からPDFの作成も標準（「Adobe Acrobat」または（Production Premium以外の）「Adobe Creative Suite」の入っているPCではそれを、入っていないPCには、6.xでは付属の「Acrobat Elements 7.0」を、7.xでは付属の「Adobe PDF Creation Add-On」をそれぞれ利用）でできる。競合規格という側面もあるが、共栄共存規格という側面も持ち合わせている。

FlashPaper

Macromedia社が策定。PDFと同様に各種文書から「印刷」して作成でき、作成された文書はレイアウトを保持したまま表示・

印刷できる。Macromedia社がAdobeに買収されたことで、CS3世代ではFlashPaperは作成できなくなり実質的にPDFに一本化された。

Scalable Vector Graphics (SVG)

各種図形を表すファイル形式であるSVGは、PDFと同じく文書のレイアウトを保持したまま表示・印刷する用途に利用できる。現状では、SVGが普及していないが、IE(8以前)を除いて、ほとんどのブラウザで利用できる。そうした中で、XSL FormatterはXML文書からPDFと同等のレイアウト結果をSVGで出力できる。

XML Paper Specification (XPS)

マイクロソフト が策定し、Windows VistaやMicrosoft Officeで採用される印刷用のプラットフォーム。PDFとよく似た機能をもち、「PDFキラー」を標榜していたが、Microsoft OfficeでPDFを直接書き出せるようになったこと、Windows 10では標準でPDFを表示できるようになったこと、他のOSやサードパーティー製ソフトの少なさから、主流となりえなかった。

外部リンク [編集]

ウィキメディア・コモンズには、**Portable Document Format**に関連するカテゴリがあります。

- Acrobat SDK 2021 - PDF Referenceの節にPDF各バージョンの仕様書へのリンクがある。
  - Document management — Portable document format — Part 1: PDF 1.7
- Adobe PDFとは？ - アドビ
- Adobe Sign（PDF署名） - Adobe Document Cloud (英語版)
- TeX Wiki "PDF の作り方"
- PDFChat: AI PDF Reader, PDF ドキュメントとチャットする
- LightPDF、ChatPDF、ChatDocなど - AIでPDFの解析と内容についてチャットするウェブツール
  - フォントの埋め込みとライセンス

表・話・編・歴	**PDFソフトウェア**	[表示]
表・話・編・歴	**Adobe**	[表示]
表・話・編・歴	**多目的オフィス文書ファイルフォーマット**	[表示]
典拠管理データベース: 国立図書館	ドイツ・イスラエル・アメリカ・チェコ	

カテゴリ: PDF | アドビ | オープンフォーマット | ベクターグラフィックス | ページ記述言語 | 電子書籍 | DTP | 文書作成 | ISO標準

最終更新 2024年5月18日 (土) 13:58 （日時は個人設定で未設定ならばUTC）。

テキストはクリエイティブ・コモンズ 表示-継承ライセンスのもとで利用できます。追加の条件が適用される場合があります。詳細については利用規約を参照してください。

プライバシー・ポリシー　ウィキペディアについて　免責事項　行動規範　開発者　統計　Cookieに関する声明　モバイルビュー