PDF Format Preservation Assessment

Document History

Date	Version	Author(s)	Circulation
10/12/2014	1.1	Paul Wheatley, Peter May, Maureen Pennock	External
25/02/2015	1.2	Peter May	External

British Library Digital Preservation Team
digitalpreservation@bl.uk

1. Introduction

This document provides a high level, non-collection specific assessment of the PDF file format with regard to preservation risks and the practicalities of preserving data in this format.

This format assessment is one of a series of assessments carried out by the British Library's Digital Preservation Team. An explanation of criteria used in this assessment is provided in italics below each heading.

1.1 Scope

This document will primarily focus on the most relevant (for digital preservation) and commonly encountered sub formats or profiles of the PDF family: PDF (versions 1 through 1.7) and PDF/A (versions A-1 through A-3). The PDF v2.0 ISO Standard is currently being drafted [1], and is therefore out of scope for this assessment.

Note that this assessment considers format issues only, and does not explore other factors essential to a preservation planning exercise, such as collection specific characteristics, that should always be considered before implementing preservation actions.

1.2 PDF Summary

PDF is a file format designed primarily to represent page based documents in a cross platform manner. PDF/A is a series of ISO standardised PDF profiles that restrict functionality with a potential for preservation risk. Ange Albertini's PDF101 document walk through provides a useful overview of the PDF document structure [2].

PDF/A was introduced with the aim of supporting long term archiving of digital documents with three versions standardised by ISO (ISO19005-1, 2 and 3). PDF/A-1b is based on PDF version 1.4 and acts as a restrictive profile that prohibits use of functionality considered problematic for long term archiving such as non-embedded fonts, JavaScript, attached files and encryption. PDF/A-1a adds additional requirements on document structure in order to simplify text extraction and accessibility (such as the use of tagged PDF and Unicode character maps). PDF/A-2 is based on PDF version 1.7 and allows functionality such as JPEG2000 compression and the attachment PDF/A files. It provides 3 levels of conformance: A – satisfies all requirements; B – encompasses requirements around visual appearance but not structural or semantic properties; and U – level B plus the requirement for all text to have Unicode equivalents. PDF/A-3 is also based on PDF version 1.7 and allows attachment of any file format. Like PDF/A-2, it provides 3 levels of conformance: A, B and U.

2. Assessment

2.1 Development Status

A summary of the development history of the format and an indication of its current status

Adobe created version 1 of PDF in 1993 basing it largely on a subset of Postscript. It continued adding new functionality[1] through seven major versions. Version 1.7 achieved standardisation as ISO 32000-1 in 2008, at which point, control of the specification passed to an ISO Committee who are responsible for producing future versions of the PDF specification [3]. PDF and its various sub-formats continue to be refined. The latest of the PDF/A ISO standards, PDF/A-3 was released in 2012.

2.2 Adoption and Usage

An impression of how widely used the file format is, with reference to use in other memory organisations and their practical experiences of working with the format

[1] As described in the Adobe Specifications section of [41]

PDF has become a ubiquitous format for exchanging electronic copies of page based documents. Duff Johnson's survey of document file-format popularity, as measured by way of Google's "filetype:" search, reveals 77% of documents were PDFs in February 2014 [4]. Johnson explains: "Born before the web to facilitate the exchange of hardcopy documents, PDF is the format people use when they need an electronic "hard copy" document. Many business, publishing and records-keeping applications require a reliable, flexible and capable analog for paper. Some love their TIFF files, but those are pictures, not documents. For the vast majority, PDF remains the only game in town".

PDF is used almost universally in academia as the format of choice for research papers, with PDF dominating the pre-print heavy institutional repositories of Further and Higher Education institutions. Hitchcock and Tarrant generated format profiles for a variety of repositories and note that: "For open access research repositories the typical profile is dominated by PDF and its variants and versions" [5].

The British Library holds large numbers of PDF files, with particularly large incidences of PDFs in the UK Web Archive, in eJournals, and where PDF is used as an access copy for digitised books or newspapers.

Andrew Jackson's research [6] reveals the large numbers of software applications used to create PDF files in the UK Web Archive - 2100 distinct software identifiers - and speculates that "the number of distinct implementations can be taken as an indicator for the maturity, stability and degree of standardisation of a particular format, although more thorough analysis across more formats would be required to confirm this". How many of these software applications created PDFs using native code rather than one of the many PDF libraries is unclear, but this remains a startling figure.

2.3 Software Support

2.3.1 Rendering Software Support

An overall impression of software support for rendering the format with reference to: typical desktop software; and current support on British Library reading room PCs

Support for viewing or rendering PDFs is good with a significant number of viewer applications that are at a reasonable level of maturity [7]. A number of open source viewers have been developed although support does lag behind revisions of the PDF standard as new functionality is added [8].

Issues

A variety of issues contribute to uncertainty over PDF rendering and the impact this may have on long term preservation. Adobe Reader's [9] tolerance in rendering invalid PDFs, the lack of effective PDF validation, the variable quality and support provided by 3rd party viewers and the quality of PDFs generated by a multitude of sub-standard PDF creating software all play a part in creating a complex and somewhat opaque picture of potential preservation risk.

Van der Knjiff notes that tolerance has been built into viewers due to a need for compatibility: "The PDF specification states that reader (or viewer) applications should be written in such a way that they simply ignore any unknown features (such as new features that did not yet exist when the reader was written). This also implies that if a (new) document contains features that are not recognised by the (old) reader, these features may not display the way they should, or they may not even display at all" [10].

Experiments in the Testbed Digitale Bewaring Project (including with PDF files) suggested that new viewers often behaved differently in terms of what was rendered than previous viewers [11].

Sheila Morrissey observed the tolerances to invalid PDFs present in Adobe Reader, and notes that even the minimal documentation for these disappeared from the PDF Specification on ISO

standardisation. She states “Older versions of the PDF specification included an appendix called 'Implementation Notes', which describes at least some of the deviations from the specification for which Acrobat reader attempts to compensate. These notes do not comprise a part of the ISO PDF 32000-1:2008 document. Further, these notes, while helpful, beg the question as to what we are to consider authoritative with respect to PDF format instances: the specification, or the behavior of the Acrobat reader application” [12].

This issue is exacerbated by the sheer complexity and range of functionality that has been added to the PDF Specification over the last two decades. Van der Knijff notes that “aside from Adobe, a myriad of companies, organisations and individuals offer applications for viewing PDF documents. Because of the complexity and feature-richness of the PDF format, many of these third-party applications do not support the full set of features defined in the PDF specification. This may also result in documents not appearing the way they were originally intended” [10].

Duff Johnson provides survey results that suggest that badly formed PDFs are a not insignificant problem in terms of scale: “In a survey of ECM industry professionals in March 2013 the PDF Association found a third [of respondents] claiming to personally encounter bad PDF files, or believed them to be commonplace. A quarter of respondents thought more than 1% of PDF files were broken in some way” [13]. Concrete research is of course difficult as validation tools are themselves unreliable (see section 2.3.2 below) but Van der Knijff provides some initial data on invalidity of PDFs in the GovDocs corpus [14]. His conclusions mainly refer to the lack of ground truth and validation effectiveness and further developments here would clearly be useful.

2.3.2 Preservation Software Support

An impression of the availability and effectiveness of software for managing and preserving instances of the file format

Format identification

Identification is supported by the usual range of format identification tools such as Unix File, Apache Tika and DROID. Both DROID and Tika provide version level identification. Identification of PDF/A variants could be considered more a matter of validation, and this is discussed below.

Validation, Conformance Checking and Detecting Preservation Risks

A number of software tools provide facilities to validate PDFs against the PDF specification (as in the case of JHOVE [15]) or against a version of the PDF/A specification. Comparative reports provide some indication of the quality of these tools, but regardless of their effectiveness it remains unclear how useful it is from a preservation perspective to make preservation judgements on the basis of validity alone. The tolerance of viewer applications to undocumented specification infringements (see Rendering Software Support above) complicates the situation.

Experiments with filtered results from validation tools for both conformance checking against a policy driven profile and for detecting specific known preservation risks are at an early stage but would benefit considerably from better validation software. Apache Preflight has been applied in experiments for this purpose, with experiments at a SPRUCE hackathon [16] and subsequent work from Van der Knijff highlighting the immaturity of Preflight but also the considerable progress in bug fixing resulting from experimentation and engagement [14]. Currently however, the knowledge gap between validation results [17] and how they might align with preservation risks remains significant.

Florida Virtual Campus (FVC) reported on the shortcomings of JHOVE PDF/A validation and have, as of August 2013, implemented validation using pdfaPilot [18]. Their workflow goes further in altering PDFs to fix issues of none compliance: “If a PDF is not identified as a PDF/A document, the FVC will convert the PDF into a PDF/A-1b document by applying fixes to the

PDF, such as embedding un-embedded fonts, converting device-dependent color spaces to device- independent ones, etc., and save the PDF/A-1b document as a normalized version"[2].

PDF/A Manager [19], part of the PDF Tron suite [20], is a commercial offering providing PDF validation used by the Kost-val [21] toolset at Kost [22]. Both pdfaPilot [23] and PDF/A Manager were assessed (along with 3-Heights [24]) by Carol Chou and Jamin Koo, with all products achieving PDF/A validation accuracy results of between 90 and 95% [25]. Other experiments also report on inconsistent validation tool results [26]. Concerns about the lack of effective PDF validation continue to the present day with PDF experts Duff Johnson and Sheila Morrissey both calling for energy to be devoted to solving this key challenge [27].

Flint [28], developed at the British Library as part of the SCAPE project, provides a promising framework for validating PDF files against institutional policies. It makes use of a number of other tools, including Apache Preflight and PDFBox, and has a focus on DRM detection (see section 2.9). As noted in its documentation, it is a "work-in-progress and hence far from being satisfactory from a domain-specific point of view, but should be a good guide for how to implement your own format-validation implementation" [28].

The issues noted above under Rendering Software clearly make visual inspection of PDF rendering challenging due to the tolerances of the viewers.

Metadata Extraction

There are a number of options for extracting metadata from PDFs, from both open source and commercial tools, such as Apache Tika [29], the NLNZ Metadata Extraction Tool [30], JHOVE [31], or 3-Heights™ PDF Extract [32].

Migration

Migration to and from PDF is supported by a considerable number of dedicated PDF focused applications, as well as more general applications that feature PDF support as secondary functions (such as Microsoft Word that supports creation of PDF and PDF/A natively). This is not surprising given the format's ubiquity and the comprehensive support provided PDF libraries [33]. A number of tools provide facilities for manipulating the content and structure of PDF files, which could be useful in a preservation setting (for example, QPDF [34]). PDF Tron is an example of one of the many toolsets that provides support for both generation of PDFs and migration from PDF to a variety of other formats.

It should be noted that support for quality checking of migrations via comparison between source and destination files remains poor. Validation of the result (or even comparison of validation before and after) may not provide sufficient confidence that significant properties have survived the migration. Experience in validating software for converting PDF to PDF/A suggests a range of complexities and subtleties that make even the assessment and the choice of software a challenge[3]. For example, Jenny Mitcham (Archaeology Data Service [35]) notes various challenges with PDF to PDF/A migration [36].

2.4 Documentation and Guidance

An indication of the availability of practical documentation or guidance with specific reference to the facilitation of any recommended actions

Various versions and sub formats of PDF have been ISO standardised and published including PDF version 1.7 (ISO 32000-1:2008) and PDF/A (ISO 19005), with copies provided by Adobe [3] along with an archive of legacy documentation [37]. As would be expected for a format used as widely as PDF, documentation and other support for PDF usage and tool development is available from a number of sources, such as the PDF Association [38].

[2] Note that the original PDF will remain unchanged and is always kept in the archive.
[3] See Validation, above.

Adobe Reader effectively operates as a de facto reference implementation for PDF, but arguably without some of the necessary openness required to become a genuine reference implementation. Duff Johnson noted: "In their attempt to ensure that even the sloppiest PDF files still worked, Adobe created a situation in which developers could (and have) used Adobe's Reader as the reference implementation for their PDF software. In 2010, there is still no alternative to Adobe Reader when it comes to validating third-party software" [39].

Julia Wolf notes the implications of what is not specified in ISO 32000: "While reading the ISO 32000-1 [PDF] document – or really any technical specification – what you really need to pay the most attention to, is what is not said. Not only is ISO 32000-1 absent of any formal language definition (BNF, etc.) but many possible glosses which can be formed that are not defined. (As it says right at the very beginning of the ISO 32000-1, there's nothing in this document that defines whether or not a PDF file is well-formed or not" [40]. Wolf concludes that it is "called Adobe Acrobat because it'll bend over backwards!" (to tolerate badly formed PDF files).

As a consequence of these ambiguities the impact on preservation activities, such as validation, is likely to be significant.

2.5 Complexity

An impression of the complexity of the format with respect to the impact this is likely to have on the British Library managing or working with content in this format. What level of expertise in the format is required to have confidence in management and preservation?

As the PDF format has been developed Adobe have introduced a considerable range of functionality that has added to the complexity of the format and the basic structure originally introduced in version 1. Wikipedia provides an overview of the functionality added in each version, such as embedded JPEG2000 filters, encryption, Universal 3D support and fill in forms [41]. Support for optimisation of PDFs so that partially downloaded files can begin to be displayed straight away (for example in a web browser) is provided by optionally structuring files in a "linear" fashion. The challenges involved in validating or merely supporting this considerable range of functionality are noted in various sections throughout this document. As Van der Knijff illustrates in a blog post exploring embedded files [42], even select parts of the PDF standards require careful consideration to inform involved analysis of preservation risks.

The complexity of the PDF standard(s) and the implications of this complexity for security considerations are highlighted by both Julia Wolf in OMG WTF PDF [40; 43], and Ange Albertini on Corkami [44]

2.6 Embedded or Attached Content

The potential for embedding or attaching files of similar or different formats, and the likely implications of this

PDF provides a variety of ways to embed content of other file types. Embedding of images (what might be referred to as "inline") is provided by Image xObjects which can utilise one of 10 filters (including JPEG2000 compression, JBIG or LZW). Inline multimedia is provided by Media Clip Objects. The attachment of files that do not need to be processed by a PDF reader (for example, for associating documents referenced from a PDF) is provided by embedded file streams[4]. File attachments are discussed in more detail below in relation to the various PDF/A standards.

PDF/A represents a profile of the PDF standard providing a restriction on certain features in an effort to support long-term archiving of digital documents. PDF/A-1 prohibits file embedding, as well as audio and video objects. It is based on PDF version 1.4 so prohibits JPEG2000 filters. PDF/A-2 is based on PDF version 1.7 and does not prohibit JPEG2000 compression (arguably

[4] Van der Knijff provides a detailed exploration of the ins and outs of embedding and attaching within PDF in [42]

a strange choice, see related JP2 assessment). PDF/A-2 allows embedded file streams but only if they are PDF/A files themselves. PDF/A-3 however removes a single but contentious restriction to the PDF/A-2 standard: any file format can be embedded as a file stream in a PDF/A-3. This may prove beneficial in some use cases, and especially from a preservation point of view where, for example, a source document for the PDF itself can be embedded.

It has been observed however that this attachment facility has the potential to be used for a variety of purposes, such as embedding additional information or even information of a more critical nature than the primary document. This leaves potentially difficult questions for an archive. Is an attached file of critical importance for preservation (despite potentially being of any format and hence potentially a significant preservation risk), merely a secondary object with optional or additional data, or, as noted above, the source data for the PDF? Even with the capability for indicating the intention of attached files, will PDF creators take the time to provide that necessary metadata? How will the creating software influence this process?

Attached files, of course, provide challenges for format identification tools so they will not be detected without deeper, format specific file parsing. The NDSA PDF/A-3 Working Group published a detailed discussion focused on these concerns [45], which considers a number of use cases and makes some sensible recommendations (such as checking received PDF/A-3s for embedded files, and treating PDF/A-3s separately from the other PDF/A varieties in terms of format preference lists and related format action plans). How evolution of PDF/A-3 pans out, how the software that is built to support it is developed, and how users will apply the tools is of course unknown at this time. Suffice to say that this remains a significant area of concern for the future [46].

Carl Wilson's experiments with embedded content in PDFs reinforces the complexity of this risk and the challenges in identifying content that can be embedded or attached in a variety of ways [47].

From PDF version 1.3, embedded JavaScript can be used to perform certain actions such as manipulating data from user filled forms, and from version 1.6 for manipulating embedded Universal 3D content. JavaScript adds an additional level of complexity for viewers and without it, functionality and appearance of pages when rendered may be inaccurate [10]. Van der Knijff notes deficiencies in detecting the presence of JavaScript in PDFs using Preflight [48].

2.7 External Dependencies

An indication of the possibility of content external to an instance of the file format that is complimentary or even essential to the intellectual content of the instance

A PDF may reference an external file in a number of different ways including: link annotations, references from stream objects, movie and sound annotations, web capture content, and reference xobjects [10]. Van der Knijff notes that “contrary to some (incorrect) popular belief, the PDF/A standards do not rule out references to external files completely. The following mechanisms for referring to external content are allowed: URI actions: these refer to Internet resources (i.e. a clickable hyperlink), GoToR actions: these refer to an external PDF file (i.e. a clickable link to a locally stored PDF)” [49].

Successful and accurate rendering of a PDF file may depend on font information external to the file. Assuming there are no copyright complications (see Legal Issues) font information can be embedded to avoid this problem and this is a requirement in PDF/A. Van der Knijff notes also that “in order to avoid any possible ambiguity about the font’s name, fonts should be subset as well as embedded” [10].

Verifying that a PDF file includes all necessary font information is a challenging task. Visual checking can be difficult as viewers will automatically substitute similar fonts where possible, rather than reporting any errors. Experiments with PDF/A validation tools show some potential but further research is required to ascertain which errors correspond with genuine risks [50]. Wilson observes that merely checking for the presence of a font is likely to be insufficient [51].

Kam Woods and Geoffrey Brown provide some insight into the frequency of font problems, albeit with a non-PDF specific scope: “We have shown that the majority (up to 79%) of digital documents obtained from a wide range of sources can be rendered accurately using fonts appearing in modern desktop environments such as the combination of Microsoft Windows and Microsoft Office. With a small amount of additional work – using information drawn from font foundries, or performing family name matches for legacy fonts or commercial fonts for which distribution has ceased – we can expect to increase this coverage to 92%.

This nevertheless leaves a large number of documents unaccounted for. Microsoft’s own search engine indexes nearly 60 million documents currently available on the web. At this level of coverage, 1.8 million documents are guaranteed to be rendered inconsistently on a typical workstation. For many of these documents, the loss of information may be negligible. It is impossible, however, to quantify this without appropriate software tools to analyze the risk to a particular collection” [52].

Van der Knijff assessed validation errors in the Govdocs corpus and also provided some insight on the frequency of this issue: “What is clear here is that the majority of failed tests is font-related. ...the results are consistent with the outcome of a 2013 survey by the PDF Association, which showed that its members see fonts as the most challenging aspect of PDF, both for processing and writing” [14].

2.8 Legal Issues

Legal impediments to the use, management or preservation of instances of the file format

Adobe holds a number of patents relating to PDF and has issued royalty free rights on a significant number of these in order to encourage uptake and 3rd party development of PDF tools [53]. A change of policy in at least the short term appears to be highly unlikely given the obvious success of this approach.

Where fonts are not embedded in a PDF there is risk of loss of appearance and possibly meaning (see below) but copyright restrictions may prevent embedding [10]. This may open up preservation risk or where a copyrighted font has been embedded risk copyright infraction for the preserving organisation. Chou and Koo note that: “There are ways to circumvent possible copyrights infringement through font substitution but some specialized fonts may prove to be difficult not only to procure but also to use in PDF/A conversion, as their makers can prohibit embedding of fonts” [25].

2.9 Technical Protection Mechanisms

Encryption, Digital Rights Management and any other technical mechanisms that might restrict usage, management or preservation of instances of the file format

The PDF standards permit PDFs to optionally be password protected or for elements of a document to be encrypted, with option to restrict a number of operations such as viewing or printing. Both present potential preservation risks, although password protection can easily be bypassed and some 3rd party viewers will not enforce password protection as a matter of course. Encryption is potentially more serious. The password required to decrypt an encrypted PDF could be cracked if deemed to be legal, but a strong password may make this a time consuming challenge, albeit one that will diminish over time as computational power increases.

2.10 Other Preservation Risks

Other evidence based preservation risks, noting that many known preservation risks are format specific and do not easily fit under any of the sustainability factors above

Johan van der Knijff notes that newer versions of PDF are backwards inclusive but not entirely so: “In principle, newer versions are always backward-inclusive; however, the ISO 32000 edition contains the following statement: 'The specifications for PDF are backward inclusive, meaning that PDF 1.7 includes all of the functionality previously documented in the Adobe PDF Specifications for versions 1.0 through 1.6. It should be noted that where Adobe removed

certain features of PDF from their standard, they too are not contained herein.' ISO 32000 does not provide any information on which features have been removed during the evolution of the format" [41].

The nature of the restrictions in PDF/A preclude preservation of some functionality. Its application will therefore not necessarily suit every use case. For example, wholesale migration of a PDF collection to one of the PDF/A versions is unwise as functionality such as audio and video will be discarded. This scenario can be considered as one specific example from the more general case of preserving significant properties when migrating a file to PDF. Under such circumstances, thought should be given as to the aspects of the original file that must be preserved and whether or not PDF (or PDF/A) can support them.

Unlike migration, receipt or deposit of a PDF/A-1 may not raise significant preservation concerns as the PDF/A restrictions prohibit functionality associated with the preservation risks identified in this report. Assuming of course that the source is trusted and the PDF/A-1 does indeed conform to the restrictions described in the PDF/A-1 standard; this is perhaps a potentially dangerous assumption and one that may be difficult to test given concerns about PDF/A validation.

2.11 Preservation Risk Summary

A summary of preservation risks and recommended actions (where possible)

PDF is a ubiquitous format in the contemporary computing world but widespread adoption, usage and software support has not led to the universal mitigation of preservation risks associated with this format.

The presence of invalid or badly formed PDF files in deposited collections appears to be highly likely but the impact on preservation and future access is unclear. Further research, study of specific collections, and analysis of validation tools would help to clarify the situation. Improved validation software would go a long way to addressing the challenge itself. Lobbying of the industry and contributions to open source validators (even if those contributions are only bug reports) could be considered as useful actions. The former of these could perhaps be pursued by digital preservation advocacy organisations. Further exploration and research on the various font related risks, primarily investigating the frequency and impact of missing font information, is arguably the other key priority.

Of the remaining risks listed below, encryption appears to be of the highest impact and, depending on collection requirements, detection of encryption could be important. For example, for PDFs deposited under legal deposit, an automated detection mechanism would allow encrypted PDFs to be rejected.

The following list summarises risks for PDF1.7 and versions prior to that where those features are supported. Most of these risks are precluded from PDF/A but given the uncertainty of support and accuracy for PDF generation and validation applications these remain potential risks for files that may in fact only purport to be PDF/A.

- **Invalid or badly formed PDF files**
  - May affect ability to render files now or in the future
- **Legal issues**
  - Embedded copyrighted fonts may pose copyright infraction for the preserving organisation
- **Missing font information**
  - Where not easily substituted, could lead to loss of critical information, particularly where, for example, mathematical formula are present
- **Encryption**
  - Password protection or encryption of elements of a document may prevent viewing

  - Optional restrictions may prevent specific kinds of use of a PDF, e.g. printing

- **File attachments**
  - Attachments of any file format could pose a variety of preservation risks in themselves
- **JavaScript and executable file launch**
  - Enable complex interactive behaviour that without it could result in inaccurate rendering
- **Embedded multimedia content**
  - Included video, sound and/or JPEG2000 streams[5] may be discarded on conversion to PDF/A
- **External References**
  - References to external files (some of which are restricted in PDF/A) may affect rendering if they change or disappear

Note also that PDF has the potential for a variety of security exploits [54].

3. Recommendations for Action

Recommended actions in usage and handling of the format. Recommend actions in the support or development of software applications that provide, or have the potential to provide, significant risk mitigation for the format. Note that these recommendations do not take into account other requirements such as those driven by specific British Library collections, or non-preservation issues such as resourcing.

The complex situation that has led to the common occurrence of invalid or badly formed PDFs (caused by poor support for validation, the tolerance of PDF viewers such as Adobe Reader, and the lack of accuracy of PDF creating applications) creates a worrying situation for those preserving PDF for the long term. The impact of this situation on long term preservation is unclear and would benefit from further research. A number of the other identified PDF features and/or risks have the potential to be catastrophic from a preservation point of view (such as encryption or missing font information). Strengthening our ability to detect these risks and ultimately developing trusted (and verifiable) means of fixing these issues in PDF files will be essential.

Handling Recommendations

It is recommended that PDFs are created to one of the PDF/A standards (ideally PDF/A-1) and ideally validated using a software application/suite different to that of the creating software itself. Although research indicates that PDF validation is imperfect it is likely to be able to catch obvious PDF functionality that should be avoided and therefore act as a useful quality assurance process as well as simplifying future preservation work.

Where PDFs are deposited with a collecting agency, it is suspected that checking conformance to a subset of a selected version of PDF/A using a PDF/A validator would be useful in identifying preservation risks. However, the severity and frequency of these risks in collections remains unknown. Modifying PDFs in order to meet preservation criteria could potentially do more damage than good, and will not be a trivial or inexpensive matter. It is therefore recommended that further research is conducted before efforts are focused on operationally assessing or validating deposited PDFs.

Software Recommendations

As noted above, the severity and frequency of the risks identified above remain relatively poorly understood. Existing published research has only begun to scratch the surface in revealing how these risks may affect an archive collection of PDF files (or not as the case may be!). Research to apply validation tools to collections in order to more clearly identify genuinely

[5] Note that JPEG2000 streams are forbidden in PDF/A-1 but not in PDF/A-2

problematic PDFs[6], or indeed discount identified risks whose frequency or impact is not significant, would help considerably to inform handling guidelines and potentially avoid overly prescriptive and potentially costly PDF fixing that has been adopted by some organisations. It is suggested that research of this kind examines a variety of tools and examines both organisation-held data and one of the publicly available corpora ensuring both relevancy and reproducibility of the work. Key targets for the work should include:

- Better understanding of the effectiveness/accuracy of existing PDF validators
- Understanding how PDF validators might be applied in identifying PDF risks (in particular, which validation reports relate to which risks)
- Understanding the frequency and impact of PDF risks in specific collections (through application of the above)

Collaborating with other organisations to encourage and support the development of an effective PDF validator may well have a significant effect on the overall quality of PDF creating applications and ultimately the PDFs which will become part of collections in the future.

Monitoring Recommendations

The preservation risks faced by PDF are unlikely to change rapidly and so review of this document should not be considered a high priority. However, awareness of new software developments that may provide useful PDF validation mechanisms should be maintained.

4. References

1. ISO/DIS 32000-2 PDF 2.0. *ISO.* [Online] [Cited: 24 February 2015.] http://www.iso.org/iso/catalogue_detail.htm?csnumber=63534.

2. **Albertini, Ange.** A PDF 101 document walk through. [Online] [Cited: 24 February 2015.] http://imgur.com/a/PbN8H#7.

3. PDF Reference and Adobe Extensions to the PDF Specification. *Adobe.* [Online] [Cited: 24 February 2015.] http://www.adobe.com/devnet/pdf/pdf_reference.html.

4. **Johnson, Duff.** The 8 most popular document formats on the Web. *Duff Johnson Strategy and Communications blog.* [Online] 17 February 2014. [Cited: 24 February 2015.] http://duff-johnson.com/2014/02/17/the-8-most-popular-document-formats-on-the-web/.

5. *Characterising and Preserving Digital Repositories: File Format Profiles.* **Hitchcock, Steve and Tarrant, David.** 66, January 2011, Ariadne. http://eprints.soton.ac.uk/273241/.

6. *Formats over Time: Exploring UK Web History.* **Jackson, Andrew N.** Toronto : s.n., 2012. iPRES 2012. http://arxiv.org/abs/1210.1714.

7. **Johnson, Duff.** PDF Readers - 5 readers compared. *Talking PDF.* [Online] 30 November 2010. [Cited: 24 February 2015.] http://talkingpdf.org/pdf-readers-5-readers-compared/.

8. List of PDF Software. *Wikipedia.* [Online] [Cited: 24 February 2015.] http://en.wikipedia.org/wiki/List_of_PDF_software.

9. Adobe Reader XI. *Adobe.* [Online] [Cited: 24 February 2015.] http://www.adobe.com/uk/products/reader.html.

10. **van der Knijff, Johan.** Adobe Portable Document Format: Inventory of long term preservation risks. *Open Preservation Foundation.* [Online] 20 October 2009. [Cited: 24 February 2015.] http://www.openplanetsfoundation.org/system/files/PDFInventoryPreservationRisks_0_2_0.pdf .

11. Migration: Context and Current Status. *Digital Preservation Testbed.* [Online] 5 December 2001. [Cited: 24 February 2015.] http://www.nationaalarchief.nl/sites/default/files/docs/kennisbank/migration_0.pdf.

[6] For example identifying missing fonts that compromise rendering of mathematical formula

12. *The Network is the Format: PDF and the Long-term Use of Digital Content.* **Morrissey, Sheila.** Copenhagen, DK : s.n., 2012. Archiving 2012. Vol. 8, pp. 200-203. http://www.portico.org/digital-preservation/wp-content/uploads/2012/12/Archiving2012TheNetworkIsTheFormat.pdf.

13. **Johnson, Duff.** Are Your Documents Readable? How Would You Know? *Duff Johnson Strategy and Communications Blog.* [Online] 24 January 2014. [Cited: 24 February 2015.] http://duff-johnson.com/2014/01/24/are-your-documents-readable-how-would-you-know/.

14. **van der Knijff, Johan.** Identification of PDF Preservation Risks: Analysis of GovDocs Selected Corpus. *OPF Blog.* [Online] 27 January 2014. [Cited: 24 February 2015.] http://www.openplanetsfoundation.org/blogs/2014-01-27-identification-pdf-preservation-risks-analysis-govdocs-selected-corpus.

15. JHove PDF-hul Module. *JHove Sourceforge.* [Online] [Cited: 24 February 2015.] http://jhove.sourceforge.net/pdf-hul.html.

16. SPRUCE Characterisation Hackathon. *OPF Wiki.* [Online] 2013. [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/SPR/SPRUCE+Hackathon+Leeds%2C+Unified+Characterisation.

17. **Cliff, Peter.** Visual Analysis of PreFlight Output. *OPF Wiki.* [Online] [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/SPR/Visual+Analysis+of+Preflight+Output.

18. PDF-A Validation and Conversion in Florida Digital Archive. *Florida Virtual Campus.* [Online] 19 September 2013. [Cited: 24 February 2015.] https://share.fcla.edu/FDAPublic/Affiliates/FDA_PDF-A_validation_conversion.pdf.

19. PDF/A Manager. *PDFTron.* [Online] [Cited: 24 February 2015.] http://www.pdftron.com/pdfamanager/index.html.

20. Tools and Utilities. *PDFTron.* [Online] [Cited: 24 February 2015.] http://www.pdftron.com/pdftools.html.

21. KOST-Val. *COPTR Digital Preservation Technical Registry.* [Online] [Cited: 24 February 2015.] http://coptr.digipres.org/KOST-Val.

22. *KOST-CECO.* [Online] [Cited: 24 February 2015.] http://kost-ceco.ch/cms/.

23. PDFA Pilot. *Callas Software.* [Online] [Cited: 24 February 2015.] http://www.callassoftware.com/callas/doku.php/en:products:pdfapilot.

24. 3-Heights(TM) PDF to PDF/A Converter. *PDF Tools.* [Online] [Cited: 24 February 2015.] http://www.pdf-tools.com/pdf/pdf-to-pdfa-converter-signature.aspx.

25. *PDF to PDF/A: Evaluation of Converter Software for Implementation in Digital Repository Workflow.* **Koo, Jamin and Chou, Carol C. H.** 2012. iPRES 2012. pp. 302-303. https://ipres.ischool.utoronto.ca/sites/ipres.ischool.utoronto.ca/files/iPres 2012 Conference Proceedings Final.pdf.

26. **Gilham, Jo and Cliff, Peter.** PDFA SPRUCE Scenario. *OPF Wiki.* [Online] 2013. [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/SPR/PDFA+Validation+tools+give+different+results.

27. **Johnson, Duff.** PDF Validation Dream or Yawn? Waking up to the possibilities of an open-source PDF validator. *Duff Johnson Strategy and Communications Blog.* [Online] 2013. [Cited: 24 February 2015.] http://duff-johnson.com/wp-content/uploads/2014/01/PDFValidationDreamOrYawn.pdf.

28. Flint. *Github.* [Online] [Cited: 24 February 2015.] http://openpreserve.github.io/flint/.

29. Apache Tika. *Apache Software Foundation.* [Online] [Cited: 24 February 2015.] http://tika.apache.org.

30. Metadata Extraction Tool. *National Library of New Zealand.* [Online] [Cited: 24 February 2015.] http://meta-extractor.sourceforge.net/.

31. JHove. *JHove Sourceforge.* [Online] [Cited: 24 February 2015.] http://sourceforge.net/projects/jhove/.

32. 3 Heights(TM) PDF Extract. *PDF-Tools.* [Online] [Cited: 24 February 2015.] http://www.pdf-tools.com/pdf/pdf-extract-content-metadata-text.aspx.

33. **Shea, Dan.** Acrobat and PDF Developer Libraries. *Planet PDF.* [Online] 13 November 2013. [Cited: 24 February 2015.] http://www.planetpdf.com/developer/article.asp?ContentID=acrobat_pdf_developer_librar&gid=6218.

34. QPDF. *QPDF Sorceforge.* [Online] [Cited: 24 February 2015.] http://qpdf.sourceforge.net/.

35. *Archaeology Data Service.* [Online] [Cited: 24 February 2015.] http://archaeologydataservice.ac.uk/.

36. **Mitcham, Jenny and Wheatley, Paul.** PDF to PDF-A conversion. *OPF Wiki.* [Online] 2012. [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/REQ/PDF+to+PDF-A+conversion.

37. Adobe PDF Reference Archives. *Adobe.* [Online] [Cited: 24 February 2015.] http://www.adobe.com/devnet/pdf/pdf_reference_archive.html.

38. *PDF Association.* [Online] [Cited: 24 February 2015.] http://www.pdfa.org/.

39. **Johnson, Duff.** Is PDF an Open Standard? *Planet PDF.* [Online] 10 June 2010. [Cited: 24 February 2015.] http://www.planetpdf.com/enterprise/article.asp?ContentID=Is_PDF_an_open_standard&page=1.

40. **Wolf, Julia.** OMG-WTF-PDF Denouement. *FireEye.* [Online] 2 February 2011. [Cited: 24 February 2015.] http://www.fireeye.com/blog/technical/cyber-exploits/2011/02/omg-wtf-pdf-denouement.html.

41. Portable Document Format. *Wikipedia.* [Online] [Cited: 24 February 2015.] http://en.wikipedia.org/wiki/Portable_Document_Format.

42. **van der Knijff, Johan.** What do we mean by "embedded" files in PDF? *OPF Blog.* [Online] 9 January 2013. [Cited: 24 February 2015.] http://www.openplanetsfoundation.org/blogs/2013-01-09-what-do-we-mean-embedded-files-pdf.

43. **Wolf, Julia.** OMG-WTF-PDF [PDF Ambiguity and Obfuscation]. *Troopers.* [Online] 31 March 2011. [Cited: 24 February 2015.] https://www.troopers.de/wp-content/uploads/2011/04/TR11_Wolf_OMG_PDF.pdf.

44. **Albertini, Ange.** PDF Tricks. *Corkami.* [Online] 2014. [Cited: 24 February 2015.] https://code.google.com/p/corkami/wiki/PDFTricks.

45. **NDSA Standards and Practices Working Group.** The Benefits and Risks of the PDF/A-3 File Format for Archival Institutions. *Digital Preservation.* [Online] February 2014. [Cited: 24 February 2015.] http://www.digitalpreservation.gov/ndsa/working_groups/documents/NDSA_PDF_A3_report_final022014.pdf.

46. **Johnson, Duff.** Archivists: No flowers for PDF/A-3. *Duff Johnson's Strategy and Communications Blog.* [Online] 28 February 2014. [Cited: 24 February 2015.] http://duff-johnson.com/2014/02/28/archivists-no-flowers-for-pdfa-3/.

47. Detect, extract and analyse embedded objects in PDFs. *OPF Wiki.* [Online] 2011. [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/AQuA/Detect%2C+extract+and+analyse+embedded+objects+in+PDFs .

48. PDF Format Issues: JavaScript . *OPF Wiki.* [Online] [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/TR/JavaScript.

49. PDF Format Issues: References to external files. *OPF Wiki.* [Online] [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/TR/References+to+external+files.

50. PDF Format Issues: Fonts missing, damaged or incomplete. *OPF Wiki.* [Online] [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/TR/Fonts+missing%2C+damaged+or+incomplete.

51. SPRUCE PDF Solution: PDF Characterisation Tool. *OPF Wiki.* [Online] 2011. [Cited: 24 February 2015.] http://wiki.opf-labs.org/display/AQuA/PDF+Characterisation+Tool.

52. *Born Broken: Fonts and Information Loss in Legacy Digital Documents.* **Brown, Geoffrey and Woods, Kam.** 1, s.l. : University of Edinburgh, 2011, International Journal of Digital Curation, Vol. 6, pp. 5-19. http://dx.doi.org/10.2218/ijdc.v6i1.168. ISSN 1746-8256.

53. **Library of Congress.** PDF (Portable Document Format) Family. *Sustainability of Digital Formats.* [Online] [Cited: 24 February 2015.] http://www.digitalpreservation.gov/formats/fdd/fdd000030.shtml.

54. PDF Current Threats. *Malware Tracker.* [Online] [Cited: 25 February 2015.] http://www.malwaretracker.com/pdfthreat.php.

55. **Owens, Evan.** Automated Workflow for the Ingest and Preservation of Electronic Journals. [Online] 2010?? http://www.portico.org/digital-preservation/wp-content/uploads/2010/01/Archiving2006-Owens-pres.pdf.

< - end - >