The Network is the Format: PDF and the Long-term Use of Digital Content

Sheila M. Morrissey; ITHAKA; Princeton, NJ USA

Abstract

The Association for Library Collections and Technical Services (ALCTS) defines the goal of digital preservation as "the accurate rendering of authenticated content over time." To that end, the preservation community has delineated a set of possible approaches (migration, emulation, digital archaeology) to ensure fidelity in rendering, and has developed converging lists of recommended formats for different uses (text, still image, sound, moving image, etc.).

Featuring prominently in many such lists is the PDF family of formats. Although it is a commercially developed format, it is widely seen as meeting many of the requirements deemed critical to reducing risk to the long-term viability of a format: it is in wide-spread use; there are many implementations, some of them open-source, of viewer applications; there is a publicly available specification of the format, control of which has been ceded to a public standards body (ISO). Further, there is ongoing work, under the ISO umbrella, to develop an "archiving" profile of PDF (ISO-190005), intended "to define a file format based on PDF, known as PDF/A, which provides a mechanism for representing electronic documents in a manner that preserves their static visual appearance over time, independent of the tools and systems used for creating, storing or rendering the files."

It is the thesis of this paper that these attributes of PDF, the admirable move by Adobe to place the specification of the format under non-commercial control, and even the developing of specifications for an archiving profile of the format, while necessary, are an insufficient warrant for the long-term usability of PDF instances. Further, the reasons why this is so clarify our understanding of what is required fully to characterize a file format. These reasons suggest that the current direction of the development of the PDF/A archiving profile may, ironically, constitute a significant departure from the warranties implicit in terming PDF/A an archival format. These reasons suggest that we not only must characterize a format instance, we must also characterize format renderers. They suggest that such anatomies of the "rendition stack" are important, not just for those who contemplate a strategy of emulation for re-enacting the original experience of an object's rendition, but also for the as-yet unanticipated uses, beyond "fidelity in rendering", of the digital objects we preserve. They put in relief the challenges of encapsulation, or even multiple possible encapsulations, of a sufficient sub-graph of the network of information about a digital object, for effective future use.

Formats and Preservation

The Association for Library Collections and Technical Services (ALCTS) defines the goal of digital preservation as "the accurate rendering of authenticated content over time." [1] To that end, the preservation community has delineated a set of possible approaches (migration, emulation, digital archaeology) to ensure fidelity in rendering in the future, and has developed converging lists of recommended formats for different uses (text, still image, sound, moving image, etc.).

Malcolm Todd, of the UK National Archives, has produced a useful synthesis of the various sets of criteria used to determine the level of risk to the long-term viability of a format [2]. The five key criteria are:

- adoption: the extent to which use of a format is widespread
- technological dependencies: whether a format depends on other technologies
- disclosure: whether file format specifications are in the public domain
- transparency: how readily a file can be identified and its contents checked
- metadata support: whether metadata is provided within the format

Featuring prominently in many lists formulated from these criteria is the PDF family of formats. Although it is a commercially developed format, it is widely seen as meeting many of the requirements deemed critical to reducing risk to the long-term viability of a format: it is in wide-spread use; there are many implementations, some of them open-source, of viewer applications that can operate on diverse platforms; there are provisions within the format for embedding metadata of various sorts; there is a publicly available specification of the format, control of which has been ceded to a public standards body (ISO). Further, there is ongoing work, under the ISO umbrella, to develop an "archiving" profile of PDF (ISO-190005), intended "to define a file format based on PDF, known as PDF/A, which provides a mechanism for representing electronic documents in a manner that preserves their static visual appearance over time, independent of the tools and systems used for creating, storing or rendering the files." [3].

Are these meliorations in what might be termed the "ecosystem" of a format a sufficient warrant for the long-term viability of instances of that format? What else might we wish to consider?

Format Characterization

Formats, as Stephen Abrams has pointed out, are key factors in the curation and management of digital assets. He defines a format as

> "a transformation from an instantiation of an information model to a tangible byte stream. This transformation can be considered in three conceptually independent stages: a semantic encoding that maps portions of the information

> model to appropriate sets of information structures; a syntactic encoding that maps these structures to a set of data units; and a serialization encoding that maps data units to sequences of bytes. A format is therefore a class defined in terms of the rules that express these three encodings." [4]

Implicit in this definition of any particular instance of a file a format as the serialization of the state of an information object is the requirement that this serialization is, so to speak, "invertible" – that the deserialization of the byte stream will reconstitute the original instantiation of the information model. This deserialization is what the National Archives of Australia have termed the "performance" of the digital object. [5]

The characterization of a digital object – its identification as an instance of a particular (version or profile) of a format, its validation against the specification of that format, and the enumeration of the significant properties of that instance (whether expressed as features of a class of formats, such as bit-mapped images, or in terms of the attributes of the format's own information model) – comprise, in terms of the OAIS information model[6], the representation information necessary to reconstitute the information object, of which the format instance constitutes the serialized state.

A successful characterization is one that that captures sufficient representation information to ensure future rendition, or performance, of a digital object. By implication, there is an original, or intended, or canonical, rendition, against which the success of the characterization -- and, by inference, preservation activity itself --can be measured.

Preservation research has therefore been occupied with attempting to determine, and if possible formalize and automate the creation of, an encapsulation of a sufficient sub-graph of information about a digital object, to ensure its effective future use. This has included research into the significant properties of categories of digital objects (still photos, audio recordings, documents).[7] It has also included analysis of the "calling stack" of the (usually original) creating/viewing application code), both to determine hardware and software dependencies for the running of that application[8], and to ferret out resource dependencies (fonts, links to other files, etc.) in the individual format instance.[9] And it has included inspection and comparison of various performance of objects in obsolescing formats, on original equipment and software, in emulation stacks of that hardware and software, and, with an implicit migration, in more current hardware and software capable of "importing" an older format instance.[10]

.

Application Characterization

The variation in rendering results observed in Mr. Cochrane's study involves digital objects that are serializations of proprietary software, for which formats there are no publicly available specifications. Conceivably, with such specifications available – as is the case with PDF – software could be constructed that could reproduce the "canonical" rendition of these objects. As Mr. Abrams says,

"In an extreme case—the complete absence of any extant PDF rendering tools—one could nevertheless fully and properly, if tediously, interpret any PDF document by reference to the published specification. [4]"

As mentioned, although PDF was originally a proprietary format, its specification has been made freely available, and has indeed become an ISO standard. Adobe's tactic of making the Acrobat viewer freely available has made the format ubiquitous. Its ubiquity in turn has led to hundreds of implementations of PDF creation and viewing applications, independent of Adobe's own Acrobat family of products.

From a preservation point of view, so far as PDF is concerned, the future is, in William Gibson's terms, already here. We are provided with the means of assessing whether meeting the five key criteria (adoption, disclosure, transparency, metadata support, and, arguably at least in the case of PDF/A, minimized external dependencies) ensures the existence of artifacts, each of which is a correct "transformation from an instantiation of an information model to a tangible byte stream".

PDF is, at least originally, a page image format. It was a developed as a platform-independent serialization of a PostScript page image file.[11] That it must always be mediated by (comparatively) sophisticated software is necessitated by the presence of various graphics operators in both image and text streams. However, those operators, along with a description of the Adobe page imaging model, and the data structures in which those operators and other attribute values of document instance are expressed, are delineated in the PDF specification. Absent such a specification, and the open-source implementation of viewer applications it has made possible, one suspects digital archeologists would be hard-put to reverse-engineer a rendition application from the (sometimes compressed and encrypted) binary contents of a PDF instance.

There are indications, however, that this specification is inadequate to describe PDF instances that are, for example, viewable in Adobe Acrobat. By implication, of course, this would indicate that such a "clean room" implementation as Mr. Abrams describes will not be sufficient to render all PDF instances that are currently viewable in Adobe's proprietary viewer (as indeed he warned might be the case). Violations of Adobe's specification that are "forgiven" by the Acrobat reader include:

- information preceding the required file header (for example, files produced on older versions of Apple's operating system, which include Apple Single and Apple Double encoding information preceding the header)
- conflicts between length value in stream dictionaries, and actual length of the stream content
- broken or missing cross-reference dictionaries
- duplicate object/generation numbers on objects in a stream
- missing terminators for streams and documents

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.

Other developers of PDF applications clearly have struggled with the gap between PDF "in theory" and PDF instances in actual practice. Comments in the PDFParser source code from Adobe's PDFBox Java application note some of the various compensatory actions taken silently to repair non-conforming documents.[12] The widely-used iText library includes an "isRebuilt()" method to indicate that it has "repaired" syntactic errors it encounters in reading and copying a source PDF file.[13]

A characterization tool such as JHOVE can note deviations from the PDF standard in a PDF instance. That characterization information can be packaged with the original object as part of its representation information. But surely the already presently expressed need, in currently used PDF applications, both to note and to develop compensatory behaviors for what seem to be common deviations from the specification, suggests that another sub-network of representation information, similar to the "graphs" of significant properties, of the hardware/software stack, and of external resources dependencies, is needed. We also need to develop the network of representation information expressing the behavior of the various PDF rendition software tools with respect to the defined information model of the format in the specification – in greater granularity of detail than simply declaring what versions or profiles of the format are supported. This is certainly true if all we are considering is a faithful recreation of the "original intent" of the document's rendition. We likely would wish to consider as well how much more true that would be for other, originally unanticipated uses of such artifacts -- as, for example, large-scale text-mining of repositories of PDF documents.

Effective Standardization

The PDF imaging model, and the data structures specified for serializations of instances of that model, are complicated enough in themselves to comprise a challenge to "clean-room" implementation. The bar is lifted higher by extensions to the original core architectural "page image" metaphor for both the format and the Adobe suite of applications which render it – extensions that include browser-like hypertext linking and multi-media facilities such as the playing of embedded or referenced audio and moving image files.

These extensions proved problematic for those who looked to PDF as a permanent, archival electronic "document of record". Concern over such features motivated the principle that informed the choices made in the specification of PDF/A-1[3] and PDF/A-2[14]: that the document instance contain within itself everything necessary (given a conforming reader) to extract the complete semantic value of the document. The archival specifications contain restrictions on fonts not completely contained within the document; on links to destinations outside the document; on the use of Reference or PostScript XObjects; and on the use of 3D, Sound, Screen, and Movie Annotation types. These specifications also stipulate that all extension schemas referenced from any metadata stream in a conforming file have their descriptions embedded in the file as well.

A major change from PDF/A-1 to PDF/A-2 was comprised in an extension that permits the embedding of other files within a PDF/A-2 document – provided that embedded file is itself PDF/A-compliant. The proposed PDF/A-3 standard extends this feature further, allowing files in *any* format to be embedded within a PDF/A-3 document. This extension was motivated by the desire to encapsulate variants of the same content in the same (PDF) container. Troubling from a preservation point of view is the lack of any requirement in the standard that a conforming reader application provide the means to extract, much less render, the embedded objects. Nor is there a well-developed requirement for expressing the relationship of those embedded files to the document content proper. This extension puts in question what is to be considered the authoritative "performance" of the contents of such a container.

Given the problematic nature of this extension, and given further the lack of a reference implementation of a PDF viewer – whether of archival or other PDF instances – it would perhaps be profitable to consider whether the considerable energies involved in producing a standard might not be better employed in constructing such a reference implementation. Barring such an undertaking, a formalization not just of the data structures, but of also of the behaviors of conforming readers and writers might facilitate the task of characterizing PDF readers and writers. While the Halting Problem [15] suggests the infeasibility of validating conforming applications, we might not unreasonably look for an open implementation of an authorized validator of PDF instances.

It is to be hoped that experience gained in articulating the characterization graph for this format will be of use in the yet more vexing task of characterizing the HTML family of formats, and the browser applications which render them.

References

[1] "Definitions of Digital Preservation | Assn. for Library Collections and Technical Services (ALCTS)", (2007), Retrieved March 22, 2012, from http://www.ala.org/alcts/resources/preserv/defdigpres0408.

[2] M. Todd, "File Formats for Preservation," Digital Preservation Coalition Technology Watch Report, 09-02 (2009). Retrieved March 22, 2012, from http://www.dpconline.org/advice/technology-watch-reports

[3] ISO 19005-1 "Document Management – Electronic document file format for long-term preservation—Part 1: Use of PDF 14. (PDF/A-1)" (2005)

[4] Stephen Abrams, "File Formats", DCC Digital Curation Manual, S.Ross, M.Day (eds), (October 2007), Retrieved March 22, 2012, from http://www.dcc.ac.uk/resource/curation-manual/chapters/file-formats

[5] H. Heslop, Davis, S. & Wilson, A., "An approach to the preservation of digital records" ,(2002) Retrieved March 22, 2012, from http://web.archive.org/web/20031217152126/http://www.naa.gov.au/recordkeeping/er/digital_preservation/Green_Paper.pdf

[6] CCSDS, Reference Model for an Open Archival Information System (OAIS). CCDS 650.0-B-1 Blue Book Issue 1 (20020

[7] Gareth Knight, "InSPECT Framework Report" (2009), Retrieved March 22, 2012, from http://www.significantproperties.org.uk/inspect-framework.pdf

[8] D. von Suchodoletz, K. Rechert, Jasper Schroder, and Jeffrey van der Hoeven, "Seven Steps for Reliable Emulation Strategies Solved Problems and Open Issues", Austrian Computer Society (OCG) 2010

[9] Andrew N. Jackson, Using Automated Dependency Analysis to Generate Representation Information, Proc. IPRES 8th International Conference on Preservation of Digital Objects, pg. 89. (2011).

[10] Euan Cochrane, "Rendering Matters: Report on the results of research into digital object rendering", Archives New Zealand (2012,

Retrieved March 22, 2012, from http://archives.govt.nz/rendering-matters-report-results-research-digital-object-rendering

[11] Warnock, J. “The Camelot Project.” PlanetPDF. (1991), Retrieved March 22, 2012, from http://www.planetpdf.com/planetpdf/pdfs/warnock_camelot.pdf

[12] Apache PDFBox Java PDF Library, Retrieved March 22, 2012, from http://pdfbox.apache.org/

[13] Bruno Lowagie, iText in Action (Manning, Greenwich, CT, 2011), pg. 162.

[14] ISO 19005-2:2011 “Document Management – Electronic document file format for long-term preservation -- Part 2: Use of ISO 32000-1 (PDF/A-2)” (2011)

[15] Alan Turing, On computable numbers, with an application to the Entscheidungsproblem, Proceedings of the London Mathematical Society, Series 2, 42, pp 230-265, (1936)

Author Biography

Sheila Morrissey is Senior Research Developer at Portico, a part of ITHAKA that preserves scholarly literature published in electronic form. Her work as a Portico partner with the California Digital Library and Stanford University Library on the next-generation JHOVE2 tool includes development of its PDF module. Her past work includes the design and development of print and electronic publishing systems, and serving as a representative to XML vocabulary standards groups.