PDF association

Application Note

PDF 2.0 Application Note 002: Associated Files

2018-10

PDF TWG

PDF Association
Friedenstr. 2A · 16321 Bernau bei Berlin · Germany

E-mail: copyright@pdfa.org
Web: pdfa.org
Published in Germany and the United States of America

Foreword

This Application Note was produced by the PDF Association's PDF Technical Working Group (TWG) to explain the use of the Associated Files feature defined in ISO 32000-2 (PDF 2.0).

This document provides interpretation of the existing specifications, and does not change the text of those specifications.

The principal authors of this document are Dietrich von Seggern (callas software), Boris Doubrov (Dual Lab) and Duff Johnson (PDF Association, ISO 32000 co-Project Leader).

Table of Contents

1 Introduction

Today, “attachments” are a widely used concept in the world of email and text messages. Every email client supports the ability to attach a file; everyone who uses a computer or smartphone learns how to use this feature. The use of attachments in email has, however, some limitations. For example, there is no interoperable way to link an attachment to a certain part of the text in the email body. Nor is it possible to specify in an interoperable way the relationship between the ‘container’ email and its attachment(s). This information - the nature (beyond MIME-type) and purpose of the attachment - can only be described in the email body.

Associated Files (ISO 32000-2, 14.13) leverage the ubiquity of PDF to build on the commonplace concept of “attachments” in electronic communications. The Associated Files model is exceptionally open, and can address a wide variety of use cases. Using the Associated Files feature, PDF writers can provide additional information about files related to a PDF file in a standardized and therefore machine-readable and potentially machine-actionable way. The Associated Files mechanism also provides for connecting metadata about a related object with the related object itself.

1.1 The history of attachments in PDF

PDF introduced attachments (embedded file streams) in PDF 1.3, in the late 1990s. PDF 1.6 introduced the optional **Desc** entry that allows for adding descriptions to any embedded file. Derived from PDF/A-3, PDF 2.0 introduces “Associated Files” to enhance interoperability by specifying the nature of the relationship between a PDF document and embedded files, as well as (optionally) the object in a container PDF file (e.g., a page, an image, or a structure element) to which an embedded file relates.

PDF 1.7 introduced the concept of “Collections” defining presentational aspects of how embedded files might be made available to users. This concept is orthogonal to the Associated Files feature.

1.2 Purpose of this document

This document provides background to the dictionaries and other entries that define Associated Files in PDF 2.0. As such, it is intended for developers who want to learn about Associated Files in PDF, and how they can improve interoperability of content beyond the exchange of digital paper.

Although the document contains typical (if generalized) use cases to demonstrate the benefits of this new feature, due to the wide range of use cases and vertical marketplaces where Associated Files could be beneficial, this document does not try to be complete in this regard.

1.3 Terms of art

For technical definitions, see ISO 32000-2.

Associated File

A file that is embedded or referenced from a PDF file using additional entries defined in ISO 32000-2, 14.13.

Attached file (or file attachment)

This term usually refers to files embedded in a PDF file, especially in the context of File Attachment annotations. When we use the term in other contexts (e.g., email), we say so.

Embedded file

A file stream that is embedded into a container PDF file.

Referenced file

An external file referenced from a PDF file. An Associated File may be either embedded or referenced (external to the PDF). Note 1 in ISO 32000-2, 14.13.2 states:

> "A file specification dictionary allows for both embedded data and referenced/external data. Both types are allowed for associated files but the embedded form is recommended."

Packaged document

Typically, a "packaged document" includes several embedded and/or associated files.

2 What are Associated Files?

Associated Files (ISO 32000-2, 14.13) enhance the concept of embedded or referenced files by standardizing a means of defining "relationships" between such files (regardless of format) and the PDF objects with which they are associated.

In PDF 2.0, a file becomes an Associated File when an **AF** entry relates it to a PDF object (e.g. the document, a page or an annotation). If this entry is present some additional requirements come into play.

Associated Files are not limited to embedded files; the concept may also be used for referenced files external to the PDF. While some of the use cases introduced in this document may be used with such referenced files, one of the main benefits of Associated Files is that they form a single entity that can be transmitted or relocated without having to make sure that such external file references are not invalidated.

Associated files are not new to PDF; the concept was introduced with PDF/A-3 in 2012. Associated files are already extensively used in internal document processes of various organizations as well as in data exchange between organizations.

3 Requirements

3.1 General

In PDF 2.0, Associated Files require:

- at least one **AF** entry in a PDF object (which can be the catalog if the **AF** entry relates to the whole PDF)
- an **AFRelationship** entry in the file specification dictionary specifying the nature of the relationship between the PDF (or a PDF object), and the related content. This entry is required to contain one of the values defined in PDF 2.0: *Source*, *Data*, *Alternative*, *Supplement*, *EncryptedPayload*, *FormData*, *Schema* or *Unspecified*. Custom values may be used where none of these entries is appropriate.

For embedded Associated Files, the following requirements also apply:

- a MIME type (as defined in RFC 8118) must be specified in the **Subtype** entry of the embedded file stream dictionary
- if the (recommended) **Params** entry is present, it shall specify the latest modification date of the embedded file.

Referenced Associated Files are permitted, but in almost all contexts, only embedded files make sense.

3.2 PDF/A-3's restrictions

PDF/A-3, which introduced Associated Files to PDF in 2012, imposes some additional requirements that are not present in PDF 2.0:

- PDF/A-3 excludes external file references, so associations can be specified only for embedded files
- Since PDF/A-3 requires the **AF** entry for all embedded files, each file's MIME-type is always required
- PDF/A-3 requires every embedded file to be an Associated File and thus requires the **AFRelationship** entry in each case
- The list of objects for which the **AF** entry may be present is explicitly open in PDF 2.0 and an **AF** entry may be used anywhere in a PDF file. However, in PDF/A-3 this list is closed and the **AF** entry may only be present in the locations explicitly identified in the standard.
- PDF/A-3 has - similar to the list defined in PDF 2.0 - a list of possible objects in which the **AF** entry may be present (ISO 19005-3, Table E.1). However, the list in PDF/A-3 lacks the **DPart** entry (since **DPart** was only defined in PDF/VT when PDF/A-3 was developed).

4 Use cases

Most current and interoperable implementations using Associated files relate the embedded or referenced file with the PDF as a whole; thus their **AF** entry appears in the document's catalog dictionary. Clause 5 describes (in general terms) a few such use cases.

Implementations can also take advantage of **AF** mechanisms to associate embedded or referenced files with specific PDF objects rather than with the whole PDF file. Clause 6 describes a few such cases.

Use cases that take advantage of the wide variety of PDF viewers present on almost all desktop computers must take into account that these viewers (at this time) typically do not display associations on the object level. However, since almost all PDF viewers already support embedded and referenced files in some fashion, Associated Files are often immediately usable. In other use cases, dedicated (PDF/A-3) solutions are appropriate, e.g. when standards or recommendations leveraging on Associated Files define further provisions to enable the desired degree of interoperability. Examples include:

- European standards for hybrid electronic invoices (Factur-X in France and ZUGFeRD in Germany) build on PDF/A-3.
- The "Drawing-free Product Documentation" specification (DFP) of the German automotive industry specifies how product documentation in PDF can be combined with a geometry representation (e.g., using the JT graphics format) in a PDF/A-3 container. Such DFP containers can be sent to suppliers as requirements specifications or for archival.

4.1 AF entry in the catalog

4.1.1 Embedding the source file

The reliability and robustness of the PDF file format limits possibilities for edits. By embedding the source file during PDF creation, both final rendering and editable source can be delivered as a package. This may be useful for drafting, e.g. for an agreement exchanged between parties, or for a spreadsheet, to provide access to the formulas used to calculate cells.

End-user applications such as Open Office already use this feature to some extent, and can create such "PDF packages" that consist of the as-intended rendered document in an open format and the editable file as a (potentially proprietary) embedded (or referenced) file. Some office applications allow users to create a PDF file with embedded original with a single mouse click, for example, in the context of exporting to PDF/A-3 files.

In this use case the **AFRelationship** entry will be *Source* and the **AF** entry will be in the document's catalog dictionary.

4.1.2 Associating machine readable data

In many important use cases - electronic invoices being the classic example - data must be readable by both humans and machines. Since invoices typically cross organizational borders, sender and receiver must agree on the format and representation of the data. A "hybrid" invoice embeds a machine-readable structure (usually XML) into a PDF (or PDF/A-3) representation of the data. Both human and machine can expect and receive useful results.

The principle is applicable to any case in which a record must be processed by machines but must also be readable by humans. An example would be of software license documentation in which product name, serial number and code can be read or printed by the human from the PDF while an embedded data structure can be read by the software, and used to activate it.

In all cases where the containing PDF and the associated file are different "renderings" of the same data, the process for its creation would typically either create the two elements (PDF and machine-readable content) in one process, or derive the PDF rendering from the structured data in the embedded or referenced file. The **AFRelationship** entry will be *Alternative* and the **AF** entry will be in the document catalog.

In a related case, discussed in clause 6, the structured data in the associated file corresponds to objects in a PDF file, not to the PDF as a whole. This data may contain measurements related to a graph or a formula, or data that is the basis for a diagram. An **AF** entry could be located in an XObject for the diagram, and the **AFRelationship** entry will have the value *Data*.

4.1.3 Creating packaged documents

PDF's model for packaged documents is an immensely powerful feature facilitating exchange of content collections while maintaining their structure. In this use case the "container" PDF will commonly have several embedded files.

Today, such file structures are usually managed in project, document or content management systems. This works well if:

- the packages do not have to be exchanged with someone who does not have access to that system, or
- the packages do not have to be migrated into another system requiring the transfer of relationships between documents from the old to the new system.

If the above conditions are not met, a PDF container file with associated files is a viable solution.

The typical alternative to a PDF file with embedded files is a ZIP container or an XML structure with embedded file streams. Neither of these approaches offers features resembling PDF's set of capabilities. A ZIP container will lose its relationship to the embedded files when unzipped; a workflow must be established in order to keep that

structure intact. An XML structure uses ASCII-7bit with a negative impact on file size, and requires specialized software to access embedded files. Nor can ZIP or XML files offer a rendering model, rich encryption options, digital signatures, interactive capabilities and other benefits.

In such cases, the **AFRelationship** entry will be *Supplement* or *Data* and the **AF** entry will usually be in the document catalog.

In all these use cases the **AFRelationship** information provided for each of the associated files makes it easier for a processor to identify related files and their relationship to the PDF. This allows for improved user interfaces, and enhances possibilities for automated processes.

4.1.4 Archiving emails

Associated Files can be used for email archival purposes. The body of the email would be converted to PDF while any email attachments would be embedded in this PDF (with much better compression than in the email). Optionally, additional Associated Files could contain the original HTML / text contents of the email body. In addition all email header fields should be converted to PDF metadata in XMP format for organizational and navigational purposes.

In a typical case, email attachments would be referenced from the document catalog; the **AFRelationship** entry will be *Source* for the original body content and *Supplement* for the embedded email attachments.

4.1.5 Encrypted payloads

In another type of PDF package, PDF 2.0 allows developers to embed an encrypted PDF or other document (a so-called *encrypted payload*) into a PDF file that serves as an *unencrypted wrapper*. The unencrypted wrapper provides guidance informing the user on how they can access the embedded encrypted payload.

In this case the encrypted payload is associated with the unencrypted wrapper PDF via the **AF** entry in the document catalog of the wrapper. The file specification dictionary for the encrypted payload includes the **AFRelationship** entry with a value of *EncryptedPayload*. In addition, the file specification dictionary also includes an encrypted payload dictionary with details of the cryptographic filter needed to decrypt the encrypted payload.

4.2 AF entry not in the catalog

4.2.1 General

Most existing applications that take advantage of Associated Files use the **AF** entry in the document catalog as the place to make the association. However, the concept of Associated Files goes well beyond association only with the file as a whole, and also allows for defining relations between embedded files and certain pages, annotations, form fields, graphics objects, structure elements in the tagging structure, DParts or any other PDF object.

4.2.2 Equations

One common use of association: a graphic object or a structure element is associated with an embedded MathML equivalent. In this case, the value of **AFRelationship** would be *Alternative*.

4.2.3 Graphs and charts

A possible use case for Associated Files is to augment content in a PDF file. If the data used to create a graph or chart is not only embedded in the PDF, but directly associated with the graph or chart, it may be leveraged by a reader to provide an augmented experience of the content. For this use, the value of **AFRelationship** would be *Data*.

4.2.4 Line art figures

Similar to equations, one may provide an alternative representation of a structure element (usually, a *Figure*) containing line art drawing as an SVG image. In this case the SVG file would be associated with the corresponding structure element via the **AF** entry in the structure element dictionary, and the value of the value of **AFRelationship** would be *Alternative.*

If the line art represents a graph or a chart as in the previous clause, one may have several Associated Files with differing values in **AFRelationship** entries (for example, *Data* for the source data used to create the chart and *Alternative* for an alternative graphical representation of the chart in SVG or other format).

4.3 PDF features adding value to Associated Files

Workflows using PDF files containing Associated Files can leverage other features of PDF to enable a variety of deliverable or process-oriented benefits.

4.3.1 Compression

In PDF, binary data does not have to be ASCII encoded (as in email or XML), and in addition every data stream (including embedded files) is compressed using ZIP algorithms.

Alternative methods for storing content along with associated material, such as email or XML, do not support compression, so PDF is a much more compact way to archive such content.

4.3.2 Encryption

PDF technology includes a variety of options to encrypt both PDF document and its embedded files with either symmetric or public-key encryption. Alternatively, one can encrypt only the document, but keep Associated Files unencrypted (this can be achieved using the *Crypt* filter with *Identity* decode parameter), see 5.5, “Encrypted payloads”.

4.3.3 Digital signatures

PDF files can be digitally signed so that any modification to the file can be immediately identified. When a PDF file with Associated Files is signed the (hash in the) signature also

covers the embedded files. This is particularly useful when the embedded file format does not allow for embedded digital signatures.

5 Risks of Associated Files

Even if an application is unaware of the Associated Files feature, rendering (and other processes) are not affected.

Associated Files build primarily on the embedded files feature, commonly supported today. The risk that a PDF processor will not be able to identify the embedded file is small, but in workflows where such PDFs are further modified processors should be tested to determine whether they are able to write out modified files without removing the Associated Files entries. A PDF writer can reduce the risk further by providing **AF** entries for Associated Files.

Relationships to *external* files are, however, not as widely used and are also less robust, as the file reference can easily be destroyed when the referenced file is moved or deleted. Their use is therefore only recommended in closed environments.

Preservation institutions should be aware that PDF files may include arbitrary embedded files. PDF/A-1 forbids embedded files and PDF/A-2 allows only PDF/A files.

PDF/A-3 minimizes risks in archival workflows by requiring that any embedded file is an Associated file associated with a limited set of objects in the PDF document. In particular, each such file has a well-defined MIME-type and a clear relationship information.

Usually, archives only accept PDF/A-3 files where they exercise some degree of control or awareness regarding the embedded files.

6 Guidance for software

6.1 PDF writers

When creating PDF files with Associated Files decisions have to be made how the file specifications that hold the Associated Files are referenced from the PDF structure. In theory it is possible to use the **AF** entry alone, however, that would limit usage to PDF processors that fully support Associated Files.

To maximize robustness and interoperability, Associated Files should be embedded (not merely referenced) and linked from the PDF structure using mechanisms defined since PDF 1.3 and which are supported by all common PDF viewers to display embedded files in their user interface. These are:

- The **EmbeddedFiles** entry in the **Names** entry in the document catalog
- File Attachment annotations

Either mechanism may be used with Associated Files. For interoperability reasons it is recommended that each embedded file use one of these mechanisms. This is important, specifically, for embedded files that are not associated with the PDF as a whole via an **AF** entry in the document catalog. Otherwise, a PDF 2.0 processor would have to traverse the whole PDF structure to determine whether there are any **AF** entries for embedded or referenced files.

For example in an open environment such as the exchange of hybrid invoices, the processing application for an invoice is in most cases not a full PDF application but could be an ERP system or accounting software. To make things easier for processors that aren't capable of interpreting the PDF per se, it is important that the PDF writer makes it as easy as possible to identify and extract the embedded XML structure. An entry in **EmbeddedFiles** names tree is the most commonly-supported way for even a minimal PDF processor to identify embedded files.

As an alternative to an entry in the **EmbeddedFiles** names tree *File Attachment* annotations may be used, and should be used if a visible annotation improves usability. If File Attachment annotations are used, an additional entry in the **EmbeddedFiles** names tree may mislead some PDF viewers into displaying two entries, so it is strongly advised to use either one or the other.

6.2 Consumer software

PDF 2.0 does not restrict where a file specification dictionary with an embedded or referenced file may occur within a PDF file. This flexibility also applies to the **AF** entry; **AF** entries containing File Specifications may also occur anywhere in a PDF file.

It is common practice to use either entries in the **Names** tree in the document catalog or File Attachment annotations, as these are the places where typical PDF viewers will search for embedded files. PDF processors attempting to identify and access embedded files should therefore analyze the **Names** tree as well as all annotation entries on all PDF pages.

Developers of indexing software for search engines should consider that a truly complete index of a PDF file would include not only the content and metadata of the packaging PDF, but also any embedded or Associated Files included in the package.

6.3 Security

Since any files may be embedded into a PDF file there is a possibility of embedded files containing malicious code. Developers of PDF consumer software should therefore make deliberate decisions about how to deal with potentially malicious embedded or referenced files. Some viewers opt to entirely preventing a user from accessing any such files, but this behavior is not prescribed in the PDF specification.

7 Bibliography

ISO 14306:2017 (JT) https://www.iso.org/standard/62770.html

ISO 16684-1:2012 (XMP) https://www.iso.org/standard/57421.html

ISO 19005-3 (PDF/A-3) https://www.iso.org/standard/57229.html

ISO 32000-2 (PDF 2.0), https://www.iso.org/standard/63534.html

Factur-X http://fnfe-mpe.org/factur-x/

ZUGFeRD http://www.ferd-net.de/news/news/index.html?changelang=4

VDA 4953-2 Drawing-free Product Documentation
https://www.vda.de/en/services/Publications/drawing-free-product-documentation.html

