

A Technical Introduction to PDF/VT



 **PDFlib** Whitepaper

The PDF/VT Standard

The PDF/VT standard has been published in 2010 as ISO 16612-2. It is »designed to enable variable document printing (VDP) in a variety of environments«. PDF/VT documents contain the final content elements and associated metadata, but not any variables or templates. The predecessor standard ISO 16612-1:2005 specified the PPML/VDX formats based on PDF 1.4. However, this older standard was missing several PDF features and relied on constructs external to PDF. The modern standard PDF/VT is based on the PDF/X-4 and PDF/X-5 standards and supports PDF 1.6 features including transparency, layers, and ICC-based color management. In addition to the requirements of PDF/X the PDF/VT standard adds supplementary features to PDF to meet the requirements of high-volume personalized printing. PDF/VT enables high-performance rendering (RIPping) of digital print files by adding efficient resource management to PDF.

Since PDF/X includes color management features and support for ICC color profiles, PDF/VT inherits the same features. While color management may present a learning curve for organizations which up to now restricted themselves to black and white printing, it opens up the arena for colorful print products with consistent color reproduction.

The scope of PDF/VT includes the important segments »variable document printing« and »transactional printing« which are defined below.

Variable Document Printing

Variable document printing (VDP) – also called personalized or one-to-one marketing – generates individualized pages, i.e. some or all printed pages are different. Variable content is taken from a database and used to customize each document. VDP is mainly used for marketing purposes since personalized documents catch the recipient's attention much faster than unpersonalized mass mailings. Some examples:

- ▶ Business correspondence with variable name and address information.
- ▶ Customized brochures with variable text and graphics elements, where the variable elements are selected based on a set of composition rules. As a result, graphically rich and unique documents are created.
- ▶ Truly personalized offerings where the actual contents vary based on the location, sex, age, purchasing history or other known information about the recipient.

Transactional Printing

Transactional printing produces records of a purchase, account, or other transaction, such as bills, statements and policies. If promotional or marketing messages are included in the transactional document the combination is called transactional-promotional, or transpromo. Common industry-specific examples:

- ▶ Financial services sector: a financial statement includes customized offers tailored to the individual recipient.

Conformance levels: PDF/VT-1, PDF/VT-2, and PDF/VT-2s

- ▶ Utility sector: an electricity or gas bill contains customized service offerings based on the actual consumption of the customer.
 - ▶ Telecommunications sector: a phone bill includes custom discount offers based on the specific type and number of phone connections the customer had in the previous month.
- ISO 16612-2 specifies three PDF/VT conformance levels, all of which are based on PDF 1.6:
- ▶ PDF/VT-1 is designed for single-file exchange and based on PDF/X-4 (PDF/X-4p is not allowed). All resources required for rendering a PDF document are contained in a self-contained PDF/VT-1 file.
 - ▶ PDF/VT-2 is designed for multi-file exchange and based on one of PDF/X-4p, PDF/X-5g, or PDF/X-5pg. PDF/VT-2 documents can reference external ICC profiles, external page contents, or both. A PDF/VT document and all its referenced PDF files and external ICC profiles are collectively called a PDF/VT-2 file set.
 - ▶ PDF/VT-2s is designed for streamed delivery where document generation and rendering can be interleaved. A PDF/VT-2s stream is a MIME package containing one or more PDF/VT-1 files or PDF/VT-2 file sets (or both flavors) plus all referenced files.

Technical Concepts in PDF/VT

This section provides an overview of the technical concepts on which PDF/VT is based.

Document Part Hierarchy

The document part (DPart) hierarchy specifies the sequence and relationship of documents or parts of documents in a PDF/VT file. In a common scenario the PDF/VT file contains sub-documents for many recipients and each document part corresponds to the pages for a single recipient. In addition to assigning pages to recipients, the document part hierarchy can also reflect more complex structures. For example, the recipients may be grouped according to the ZIP code in their address, the ZIP codes may be organized according to the state, the states according to the countries, etc. This kind of document organization creates a tree-like structure which includes all pages in the document. The elements of this tree are called DPart nodes, where each internal node contains other DPart nodes and each leaf node specifies one or more pages for a recipient.

The document part hierarchy in a PDF/VT file can be used for accessing pages, alternatively to other methods such as access by page number or by page label. The DPart hierarchy is required in PDF/VT files. The optional *RecordLevel* value selects the level in the DPart hierarchy which corresponds to the records for individual recipients. This is relevant for the scope hints (see below). The DPart hierarchy will also be included in PDF 2.0 according to ISO 32000-2.

Document Part Metadata

Each node in the document part hierarchy from the root down to the leaves in the document tree can contain Document Part Metadata (DPM). It can be used to communicate information about a particular recipient's sub-document and its parts. In particular, properties which are relevant for production (e.g. number of copies of a document part) or information about the recipient (e.g. the corresponding ZIP code) can be encoded in DPM.

The PDF/VT standard specifies general means for storing document part metadata, but does not prescribe any specific type of metadata scheme or encoding. However, the standard has been developed with a bias towards the metadata scheme which is part of the Job Definition Format (JDF), a job ticket format standardized by the *International Cooperation for the Integration of Processes in Prepress, Press, and Postpress Organization* (CIP4). The metadata format recommended for use with PDF/VT is called »ICS – Common Metadata for Document Production Workflows«. For more information please visit www.cip4.org.

JDF (or other) production metadata is not required in PDF/VT, but adds substantial value in JDF-enabled workflows. The PDF/VT standard also specifies a method for representing document part metadata as an external XML document.

Optimizations for recurring Graphical Content

Print elements are often re-used on multiple pages, e.g. a company logo or product image which appears on all pages of a mailing. Optimized processing of recurring graphical content is an important strategy for improving file size and processing speed of print files. PDF has always supported XObjects as a means for optimizing the file size by including the required data for a print element only once in the file, and allowing references to this data from multiple pages (or multiple instances on the same page). XObjects may contain a raster image or arbitrary text and vector graphics contents. While XObjects in PDF aim at optimizing the overall size of a document, PDF up to now did not include any means for optimized rendering of repeated page contents. There is nothing in PDF which could tell the

consuming software that, say, an image on a particular page will appear again on another page later in the same document or maybe in the next print job. PDF/VT extends the existing concept of XObjects in PDF and adds the following means for optimizing print performance:

- ▶ **Unique identification:** XObjects can be assigned an identifier (called *GTS_XID*) which is unique across all documents. This identifier can be used by caching implementations which need to identify equivalent XObjects. In simple terms, the graphic which has already been processed for job 1 and is found to be re-used in job 2 does not have to be ripped again, but the rasterized results can be taken from the cache.
- ▶ **Scope hints and environment context:** XObjects may contain information (called *GTS_Scope*) regarding the range of pages or documents where the graphical content is re-used. This way XObjects can carry information about the useful lifetime of their rendered results in the cache: will the content be re-used only for the current recipient, re-used anywhere else in the same file or file stream, or not at all. If an environment context (called *GTS_Env*) is provided, the XObject may specify global use, i.e. it will be re-used in more than one PDF/VT instance. There are no restrictions regarding the environment context string. For example, a customer or job name can be used for identifying the environment.
- ▶ **Encapsulation hints:** XObject caching algorithms must take into account the interaction of an XObject with the calling context and existing print elements on the same page (or other pages, e.g. when imposing multiple pages on the same sheet). For example, if an XObject does not specify the color or line width, but varies its appearance based on the color and line width in effect when it is referenced, caching of the rendered result is ineffective due to the varying appearance. A similar situation arises if the XObject contains transparent elements so that the existing background must be blended with the XObject. In order to facilitate XObject caching, PDF/VT introduces the concept of Encapsulated XObjects which can be marked as such (using the *GTS_Encapsulated* key). Encapsulated XObjects must satisfy certain rules which facilitate caching.

All of these entries are optional: PDF/VT does not require any of the optimizations for recurring graphical content, but using them offers significant print performance advantages with PDF/VT capable RIPs.

Displaying and Printing PDF/VT Documents

At the core PDF/VT documents are plain PDF files plus certain additional features. However, the PDF/VT-specific additions do not affect page rendering: the pages of a PDF/VT document can be viewed with a plain PDF viewer or rendered with a PDF capable RIP which may not directly support PDF/VT, e.g. Acrobat or a third-party PDF viewer or RIP. However, some aspects must be treated carefully:

- ▶ The color management features in PDF/VT require a PDF viewer which fully implements color management with ICC profiles. This parallels the existing PDF/A and PDF/X standards which also mandate color management. Modern viewers such as Acrobat 9 and above fully support the color-related requirements for PDF/VT viewing.
- ▶ The document part hierarchy, Document Part Metadata (DPM) and optimizations for recurring graphical content won't have any effect in a plain PDF viewer.
- ▶ PDF/VT-2 allows the use of externally referenced page content. This feature is only available in Acrobat 9 and above. In addition, external references require careful configuration of the viewer.
- ▶ PDF/VT-2s packages are not PDF documents. The contents of a PDF/VT-2s MIME package must be unpacked before the documents can be viewed with a plain PDF viewer.

Common Misconceptions regarding PDF/VT

In various publications some misinformation about PDF/VT has been spread. Let's get the following points straight:

- ▶ Despite its name, PDF/VT does not include any variables: all pages contain or reference the final content, without any headroom for dynamic content creation or document formatting. Similarly, the claim that Acrobat displays PDF/VT documents except for the variable data parts is wrong.
- ▶ While PDF/VT-2s is designed for streaming PDF generation, creating a PDF/VT document cannot be interleaved with consuming (printing) it since PDF documents (PDF/VT or other) must be completely available before rendering can start. The only exception to this rule is linearized PDF for Web delivery, but this technique is optimized for fast rendering of the first PDF page in the browser, not for printing thousands of pages while the remainder of the document is still being created. A scenario which indeed is possible with PDF/VT is the following: a PDF/VT-2s package contains

multiple documents which are streamed over the network in a MIME package. As soon as all required chunks for a document within the package are available, rendering can start for this document, while other documents in the same MIME package are still transferred over the network.

PDF/VT and PDF/A for Archiving

PDF/VT-1 documents can at the same time conform to PDF/A for archiving (ISO 19005). This means that the same document which is sent to the digital printer can be archived in PDF/A format (PDF/VT-2 can not be combined with PDF/A because external references are not allowed in PDF/A). Since PDF/A-1 is based on PDF 1.4, however, combined PDF/VT + PDF/A documents must restrict themselves to the set of PDF features available in PDF 1.4. In particular, transparency and layers can not be used. The situation will change with PDF/A-2 which is based on ISO 32 000, i.e. PDF 1.7.

PDFlib and PDF/VT

PDF/VT support in PDFlib is based on PDF/X-4 and PDF/X-5 support which has been introduced in 2009. All relevant PDF/X features and requirements in PDFlib therefore apply to PDF/VT as well. This includes color management features with ICC profiles, output intents for characterizing the intended printing condition, and rules regarding the use of color, layers, and transparency in the document. Since PDFlib already supported externally referenced document components for the PDF/X-4p, PDF/X-5g and PDF/X-5pg standards it provided a good basis for the PDF/VT-2 implementation. Similarly, since PDFlib has always supported in-memory document generation and is one of the very few products which can interleave PDF creation and output the basis for PDF/VT-2s was also already available. PDFlib supports the creation of documents which conform to PDF/VT and PDF/A at the same time.

PDFlib features for creating PDF/VT-1 and PDF/VT-2 have been added with a focus on automatic creation of unique identifiers and a flexible API for describing document part hierarchy and Document Part Metadata (DPM). Since PDFlib doesn't know anything about the relationship of multiple documents which have been created separately, scope hints and encapsulation hints can be provided by the application developer. Programming examples for creating PDF/VT are available in the PDFlib Cookbook on www.pdflib.com.

PDF/VT-2s creation can be implemented on top of PDF/VT-1 or PDF/VT-2 by wrapping the PDF/VT documents and all referenced (external) ICC profiles and PDF contents in a MIME package. The rules for creating standard-conforming PDF/VT-2s MIME packages are described in the PDFlib documentation. Sample code for this task is also available in the PDFlib Cookbook.

The PDFlib pCOS Cookbook includes a code sample for retrieving DPM from a PDF/VT document with the pCOS interface and creating the corresponding XML representation as specified in the standard.

Availability

PDFlib 8 VT Edition (or PDFlib/VT for short) includes full support for creating PDF/VT output. Developers interested in creating PDF/VT output with PDFlib can request an evaluation version of PDFlib 8 VT Edition. Licensing details are negotiated on an individual basis. A PDFlib 8.0 license does not cover the use of PDFlib 8 VT Edition.



PDFlib GmbH

Franziska-Bilek-Weg 9
80339 München, Germany
phone +49 • 89 • 452 33 84-0
support@pdflib.com
www.pdflib.com/knowledge-base/pdfvt

PDFlib GmbH is completely focused on PDF technology. Customers worldwide use PDFlib products since 1997. The company closely follows development and market trends, such as ISO standards for PDF. PDFlib GmbH products are distributed all over the world with major markets in North America, Europe, and Japan.