

**ISO/TC 130/WG2/TF3 — Variable Data Printing  
Minutes of 5<sup>th</sup> Meeting  
Paris, France  
14-15 April 2008**

**1 Call to Order - Welcome - Introductions, Roll Call of Experts, Announcements**

The meeting was called to order at 9:00 a.m. by Tim Donahue, Chair. The following technical experts were in attendance and introduced themselves:

|                        |                                 |         |
|------------------------|---------------------------------|---------|
| Tim Donahue, Chair TF3 | Eastman Kodak Company           | USA     |
| Mary Abbott, Secretary | NPES                            | USA     |
| Martin Bailey          | Global Graphics                 | UK      |
| Ray Cheydleur          | X-Rite                          | USA     |
| Olaf Drümmer           | Calls Software                  | Germany |
| Tony Hittema           | AFNOR                           | France  |
| Jyri Hujanen           | Hansaprint Ltd.                 | Finland |
| Dov Isaacs             | Adobe                           | USA     |
| Akihiro Ito            | Fuji Xerox                      | Japan   |
| Andy Kraushaar         | Fogra                           | Germany |
| Makoto Matsuki         | NTT Quaris                      | Japan   |
| David McDowell         | NPES                            | USA     |
| Bruno Mortara          | ABTG                            | Brazil  |
| Rainer Prosi           | Heidelberg                      | Germany |
| Craig Revie            | Fujifilm                        | UK      |
| Leonard Rosenthol      | Adobe                           | USA     |
| Reuven Sherwin         | XMPie                           | USA     |
| Steve Smiley           | Vertis/FTA                      | USA     |
| Johnny Sutton          | Quebecor World                  | USA     |
| Yuken Tanabe           | Epson UK, Ltd.                  | UK      |
| Genji Tao              | JPMA                            | Japan   |
| Hitoshi Urabe          | Fujifilm                        | Japan   |
| David Williams         | Ricoh Americas                  | USA     |
| Dan Wilson             | Prepress I.T.                   | Ireland |
| Tomonori Yuasa         | Muroran Institute of Technology | Japan   |

The attendance sheet is WG2/TF3 N 067.

**2. Review and approve agenda (WG2/TF3 N 059R)**

The committee reviewed the draft agenda, which had been distributed prior to the meeting. There being no changes, the agenda was approved as distributed.

**3. Approve minutes of 25 September 2007 meeting (N 043)**

The committee reviewed the minutes of the 25 September 2007 meeting, which had been distributed by e-mail. There being no corrections, the minutes were approved as distributed.

#### 4. Report on interim teleconference meetings

Since the September 2007 meeting, five teleconference meeting have been held. The following are some of the issues discussed and decisions made during those meetings.

##### 4.1 Proposed Nickname: PDF/VT

There was discussion to determine a "nickname" for this file format for easy reference. It was agreed to following the style of the PDF/X and other PDF-related standards, and used PDF/VT. In this case, it was agreed that VT standards for "variable" and "transactional" data.

##### 4.2 Page eXtension Metadata (PXM)

Metadata stored within a file was formerly referred to as PPM (Page Production Metadata). To avoid possible confusion with PPML, it was agreed that it will now be called Page eXtension Metadata (PXM).

##### 4.3 Conformance levels: PDF/VT-1 and PDF/VT-2

It was agreed that two conformance levels are desirable and are defined as follows:

**PDF/VT-1** is the conformance level designation for a single file instance where all resources necessary to properly rendering the file must be included in the file. This is the conformance level that supports complete exchange. A PDF/VT-1 file shall therefore have no external dependencies and its conformance is based on PDF/X-4.

**PDF/VT-2** is the conformance level designation for a multi-file instance that permits external references to auxiliary page content defined as pages of other PDF files. The ICC profile files referenced from OutputIntent dictionaries may also be stored as external files. The PDF/VT-2 file is based on PDF/X-5g, which allows reference to externally defined PDF pages, and on PDF/X-5pg which also allows references to external ICC profiles.

It was questioned whether or not the conformance levels should be included in the title of the standard rather than just publish the root nickname of "PDF/VT" in the document title. The thinking is that it would make it easier to identify the standard if it included the conformance level nickname designation so that it would be more visible in an on-line search engine that may reference this document. It was agreed that "PDF/VT" should be changed to "PDF/VT-1 and PDF/VT-2" in the standard of the title.

The title of the document as agreed to by the committee now reads:

Graphic technology — Variable data exchange — Part 2: Using PDF/X-4 and PDF/X-5  
(PDF/VT-1 and PDF/VT-2)

**Action Item 08-01:** McDowell will notify the TC 130 Secretariat to request that ISO change the title of the work item for ISO 16612-2 in their records.

#### 4.4 Alternative record structure proposal (VTStructRoot and VTStruct)

It was discussed that a new data structure known as the *VTStruct tree* had been proposed, reviewed and adopted in earlier TF3 teleconferences prior to this meeting. This new data structure is intended to replace the original proposed use of the PDF Outline dictionary data structure as the means for defining the record and sub-document structure of a PDF/VT file instance. The new *VTStruct tree* data structure is used for specifying both variable record and sub-document structure as well as for specifying PXM metadata within the structuring context.

Details of the *VTStruct tree* data structure and existing open issues were covered during discussion later in the meeting.

#### 5. Discussion of open issues identified in interim teleconference meetings:

Matsuki presented comments from Japan relating to the proposed VTStruct data structure alternative to the Outline dictionary adopted in an interim WG2/TF3 meeting. His presentation is document N 066.

One question raised was why create a new VTStruct data structure to define record and document structure and carry the metadata rather than leverage the existing Outline dictionary structure as originally proposed.

It was noted that although the committee had first considered using the Outline tree as the basis for specifying record and sub-document structure, it was felt by some that extending it to represent the record and sub-document structure overloads the original intent of the Outline tree for a purpose other than what it was originally intended. It was felt that it would be less problematic to instead add a new data structure designed specifically for variable document applications, namely the VTStructRoot and VTStruct dictionaries.

As VTStruct is currently defined, it was argued that the Outline Dictionary would only require minor extensions in order to achieve the desired functionality. It was pointed out, however, that over time as more experience is gained with PDF/VT in practice there is likelihood that future capability will be added to this data structure to support of capabilities not yet considered (e.g. future versions of the PDF/VT standard). Such features may extend beyond the fundamental purpose of the Outline dictionary and present a problem later on. The new VTStruct data structure, on the other hand, provides us with a dedicated data structure specific to VDP applications, which can be extended in a way that doesn't alter the general and established use of the Outline dictionary.

One suggestion was that a PDF/VT file could contain both an Outline tree and a VTStruct tree where the Outline tree could provide an interactive reader view of the pages of the document if desired. Another suggestion was the use of a tool that could generate an Outline dictionary structure for a file derived from the VTStruct data structure present in the same file. An Outline dictionary in this way would enable older PDF viewers that are not VTStruct aware to provide a reasonable record oriented PDF/VT file preview functionality, among other possibly application centric possibilities.

It was pointed out that the current draft document (N 063) prohibits the presence of both the VTStructRoot and Outline dictionary in the Catalog dictionary. It was agreed that this requirement should be relaxed in the next draft document and that both data structures should be allowed to be present in a conforming PDF/VT file.

On the point that use of the Outline dictionary would provide more ready backwards compatibility with existing viewer applications already on the market, it was discussed that current viewers do not already support features of PDF/X-4 and PDF/X-5 and would need to be updated for compliance regardless of the need to make changes to support VTStruct in PDF/VT. To this end, if the file contains a VTStruct tree, then a conforming viewer should present a specific type of interface based on VTStruct.

In addition, Japan noted that there needs to be collaboration with ISO TC 171 as they continue their development of ISO 32000-2. Because in the business sector addressed by TC 171 many companies use document scanners to scan many kinds of application forms to convert their content to electronic data. This scanned data includes text and they use OCR techniques to change the scanned text image to character codes. However OCR is not always perfect and it is necessary to check the data manually using an interactive viewer.

Masaki suggested that the Outline tree or VTStruct tree defined in ISO 16612-2 could be used in a way that helps automate this comparison. The process for checking involves side-by-side comparison of the scanned image with the rendition of the OCR converted version. In this visual comparison, content errors can be more easily detected. The Outline or VTStruct approach is seen as a very useful feature for this application. For this reason, they would like to see VTStruct included in ISO 32000. They requested that if TC130/WG2/TF3 agrees with this proposal from the Japan NB of ISO/TC171, cooperation between TC171 WG7 and TC 130 is needed, perhaps in the form of a JWG. It was noted that there is a June deadline to make new proposals for consideration for ISO 32000-2.

As a result of discussion, it was tentatively agreed if a linked list data structure is appropriate for representing the VTStruct node data, it would be possible to have the VTStruct key in the Catalog point to the Outline tree, as well, thereby allowing the sharing of both Outline and VTStruct data.

Another option suggested was to add extensions into the Outline tree and add an optional Outline2 (rather than VTStruct) to point to an optional second Outline tree.

It was proposed to extend Outline dictionary to contain references to VTStruct nodes (the pages in the VTStruct), and not co-mingle them. This would be done by adding a new key to the Outline dictionary similar in concept to Dest that would instead reference VTStruct node entries. The concept of this proposal is to preserve the Outline tree as a data structure dedicated for use as an interactive electronic document page presentation mechanism as it is currently used relative to the Pages tree of any PDF file. It was suggested that the VTStruct hierarchy is architecturally more analogous to the PDF file's Pages tree and that the JDF job ticket data is more analogous to the PDF Outline dictionary since the JDF and the Outline dictionary are both used as PDF page presentation mechanisms.

**Action Item 08-02:** Donahue and Rosenthol will prepare a proposal relating to VTStruct nodes to the Outline dictionary and submit to TC 171/SC 2/ WG 7 for consideration for ISO 32000-2.

The committee looked at the requirements currently contained in 6.4 (VTStructRoot and VTStruct dictionary requirements) and made several changes. It was agreed that the set of Page dictionaries specified by the /Start and /End keys of a VTStruct dictionary entry shall not overlap the page range specified by another VTStruct dictionary entry. A requirement will also be added that the pages must be contiguous.

A note was added to the definition of "recipient record" (3.23) to clarify the fact that a particular "record" may not target an individual, but rather, a group of recipients. The new note states "Where this data structure is used for applications where specific recipients are not specified, a similar data structure is used and 'recipient' refers to an equivalent grouping of records."

It was proposed that since a single record may define documents that target a group of recipients that it is necessary to also specify a quantity of such record definitions where this quantity is written by the composition engine based on database information. It was mentioned that such information interpreted as copy count relates more to process parameterization and may be production information that belongs in the job ticket. It was then argued that in this particular case it is not really the number of copies, but rather, a short hand for the number of equivalent record instances represented by the VTStruct node.

Given this requirement, it was proposed and agreed that the copy value should be encoded as a PXM metadata key and that the PXM Key should be standardized in the document.

It was agreed to make the following changes:

- a) Define an optional PXM key to specify the number of copies of the record desired.
- b) Add a standardized PXM key for representing /RecordCount.

Changes were made to the value of /*Count* in Table 4, including the addition of a requirement that the value of the /*Count* key shall be greater than zero. It was agreed that allowing use of the negative count value for the VTStructRoot is not useful.

An informative note was added to /*RecordLevel* in Table 4 to state: "Record level zero indicates the immediate descendent VTStruct dictionary of the VTStructRoot dictionary."

/Segmentindex key was removed from Table 4. This key was considered job management level information and is unnecessary.

Table 5:

Changed to require that the VTStruct dictionary *Title* key shall always be present.

For table entries that define normative requirements, use [Required] and [Optional], or expanded variants, rather than "shall be present", etc. throughout the document.

Changed "VTStructRoot" to "VTStruct" in the value for both *Prev* and *Next*.

Use of /*Start* and /*End* in VTStruct differs from the Outline dictionary. It was suggested that /*Dest* could be used instead of /*Start* with the restriction of allowing only array; or allow both with some commonality of behavior required. After discussion, it was decided to leave it as /*Start* for now.

There appear to be many ways to address the range of pages referenced from a VTStruct leaf node, and the committee had a lengthy discussion on this matter and no agreement was reached.

Currently *Start* and *End* are defined as direct COS references to *Page* dictionaries which advantages late stage editing.

Use of */Start* and */Count* was proposed where */Start* remains a direct COS reference to the first Page of the range, and *Count* is the number of pages. It was pointed out that this approach makes file updating for late stage editing more complex.

It was decided that of */Start* and */End* in *VTStruct* will be used and referenced page ranges across leaf nodes shall not overlap and the pages in the range shall be contiguous relative to their ordering in the PDF file's Pages tree.

It was suggested that the semantics of *NodeName* be redefined as a standardized PXM key.

To clarify understanding as to the proposed use of *NodeName* it was decided that the agenda item to review JDF examples be moved forward. The committee looked at examples of PDF/VT pseudo code representing the PDF/VT *VTStruct* tree structure (N 071) and a corresponding JDF example that made use of it.

The example represented a multiple record PDF/VT instance using a non-PDF representative syntax where each recipient record defines single page cover letter and pages intended to be produced as a variable length brochure. It was shown how the structure of the *VTStruct* hierarchy is used as the interface by which a JDF based imposition processor consumes the records in sequence.

The JDF *RunList* resource is the JDF process definition interface to the PDF/VT file. The JDF *RunListLink* resource links the *RunList* resource and the PDF data to the JDF page processor and selects a page pool (*VTStruct* leaf node) sequence from the JDF *RunList*. The *Layout* resource defines mapping of pages to a sheet sequence to be printed.

In the example, JDF *Layout* resource has two sub-sections (known as partitions in JDF parlance). The first *Layout* partition is a definition that maps cover letter page to a single sided cover letter sheet and the second *Layout* partition maps the pages of the variable length brochure to a sequence of two sided one-up sheets.

It was noted that the *RunList/MetaDataMap/Expr/NameEvaluation/@Path* attribute is an XML XPath expression string that refers into the *VTStruct* hierarchy to uniquely identify metadata entries present in the various nested nodes of the structured PDL data. XPath is based on XML where XML syntax defines a hierarchy of named nodes. An equivalent hierarchy of named nodes is defined in PDF/VT using *VTStruct* where the element name corresponds to the value of the *VTStruct /NodeName* key.

Also described was the use of the JDF *RunListLink* resource in the example used as the means for sub-setting, or filtering the sub documents of records to be processed by the imposition engine.

It was agreed that the 2<sup>nd</sup> paragraph of the value description for *NodeName* (Table 5) needs to be edited to clarify the meaning of "at or above".

It was agreed that an informative annex containing a JDF + PDF/VT example will be added to the document.

It was proposed that a new */NodeNameList* key be added to the *VTStructRoot* dictionary that would define the hierarchy of node names beginning with the first level of the *VTStruct* hierarchy immediately below the *VTStructRoot*. The first entry is the name of the first level nodes of the hierarchy; the second entry is the name of the second level, and so on. This list would only define the node names down to and including the record level nodes of the hierarchy since it is already agreed that the Node names must be consistent from the root node to the record level nodes.

It was questioned whether node name symmetry needs to be maintained only above and including the record level, or perhaps throughout the entire hierarchy. As currently defined, to do a join multiple files, the hierarchy above the record level must match. Below the record level, each node level of the hierarchy could have a different *NodeName*. However, the committee also gave consideration to allowing arbitrary *NodeName* above the record level. The following proposals were considered:

Proposal 1: Relax the requirement that Records must occur at the same level – flexible for the entire structure.

Proposal 2: Define a fixed name hierarchy throughout and use PXM metadata to differentiate various products and sub-document parts.

Proposal 3: Proposal 2 and require pages to be defined in leaf nodes at the same level.

The committee agreed to Proposal 2 where use of node names across hierarchy levels must be the same throughout the hierarchy and that the hierarchy leaf nodes need not occur at the same level. The editor will revise the document to reflect this decision.

It was agreed that *NodeName* should be removed from *VTStruct*, but the content of the notes was preserved under a new entry called *NodeNameList* for later editing. The new key */NodeNameList* will be added to *VTStructRoot*.

It was agreed that the editor will also add normative text that indicates the use of the name "*VTStructRoot*" as the logical XML root element for XPath productions.

## **5.1 Optimization of *VTStructRoot* and *VTStruct* data structure organization for efficient reader processing**

It was noted that if the *VTStruct* data structures are randomly scattered throughout a PDF/VT file, it is more time-consuming to search for them for processing. It was questioned if we should require all *VTStruct* dictionaries to be located in a single, contiguous area of the file, which would make processing much more efficient for readers. It was felt that in some cases requiring this may place an undo burden on the PDF/VT generator. It was then suggested that more than one contiguous grouping of *VTStruct* dictionaries be allowed.

It was agreed to make this a recommendation in the document, but not a requirement. In this case, it also needs to be stated that the reader should not assume that the *VTStruct* dictionaries are contiguous.

**Action Item 08-03:** Rosenthol will develop a draft proposal as to how in PDF multiple VTStruct dictionaries may be specified in a contiguous section of a PDF/VT file.

## **5.2 Need for a separate conformance level requiring linear organization of PDF/VT data that allows for streaming consumption by a conforming reader**

The notion of streaming discussed is the ability to parse and process PDF/VT data as it is being downloaded to the reader.

It was discussed that streaming would require all referenced PDF resources to be defined ahead of referencing pages and the PDF pages must be defined in the order of their presentation by the consuming reader. The existing method of distributing the PDF XRef table in the file as already defined for PDF could be used. This may require some additional extensions to PDF.

It was pointed out that given the bandwidth capabilities of modern networking and storage technology and the speed at which large PDF/VT files can be transferred, the additional burden on the producer application and the added format complexity; it was agreed that streaming is something that will not be addressed by this standard.

## **5.3 Inclusion of recurrence hint information to enable further RIP efficiency**

The third issue raised by Japan relates efficient consumption of PDF/VT data by a RIP. Japan questioned why sections 6.7 (Manifest of dependent element and metadata files) and 6.8 (Recurring XObject metadata) in draft #5 were removed and do not appear in draft #6.

The committee discussed the issue of processing recurring objects efficiently. It was noted that the structure of PDF with its various types of XObject definitions allows all common source content objects to be defined a single time as uniquely identified XObjects. Given this structure, there is opportunity to build implementations that support a caching mechanism that would improve performance.

Also, an implementation can perform caching can at more than one level; for example, at the PDF parsing and at the rendering level. There are straight-forward methods for determining when/if an object should be cached that takes advantage of the existing structure of PDF, however, it was suggested that adding features to enable a particular methods of cache hinting can be problematic. It was noted that the order of consumption of the PDF/VT pages may not be the order defined in the PDF/VT file so any hinting that is order dependent may not be useful. An historical example was given that relates to cache optimization strategies in early virtual memory based computer operating systems. In the example cited, it was found that simple LRU caching tends to be most efficient and robust to differences in applications requirements, whereas more strategic caching schemes tend to have more performance failure modes for some applications. Because PDF/VT is attempting to be system-independent, it is difficult to predict what objects will need to be reused and when. Therefore, it is best that the standard not make an explicit provision for this and instead allow optimized consumers rely on the reuse meta information already present in the structure of the PDF data.

It was agreed that this explanation should be included in the document as informative material.

**Action Item 08-04:** Isaacs will prepare draft informative text relating to the decision not to permit caching/reusing objects to be included in the draft document.

#### **5.4 Need for multiple OutputIntent dictionary support in ISO 16612-2 and possible alternatives**

The committee discussed whether all parts of a PDF/VT file must be prepared for a single reference printing condition. For example, if there is to be a glossy cover, but the inside pages use a print substrate with a smaller color gamut, for which single printing condition should the PDF/VT file (set) be prepared?

The committee was asked to consider the possibility of allowing a page-specific OutputIntent. To accomplish this, a reference to an OutputIntent dictionary from the PDF Page would be necessary. A concern was then raised that allowing multiple OutputIntent dictionaries in this way would violate technical requirements for both PDF/X-4 and PDF/X-5 and compatibility with conforming readers.

Another proposal is to require the file creator to choose a single printing condition as the exchange space and the receiver must then repurpose the color data appropriately for the printer setup so as to preserve the intended appearance of the target output(s). Although this method is not ideal, it is thought to be a good "first step."

Another suggestion was the use of the new PDF Collections feature available in ISO 32000-1 which allows multiple different PDF files (e.g. PDF/X files) each prepared for a unique printing condition to be grouped into a single containing PDF file. It was concluded that this approach based on ISO 32000-1 would allow for multiple output intents, however it would be more difficult for the consumer of the PDF/VT file to process efficiently. In this approach, identification of reused XObjects must be possible across PDF files, which adds additional complexity. It was pointed out that this approach where the PDF/VT file is logically segmented among a sequence of individual PDF files (each with their own VTStruct tree) may serve to enable streaming consumption.

It was mentioned that in some cases, differences in OutputIntent may not be due to the need to print pages of the same file on different media. The desire may be to use the OutputIntent as a way of controlling the rendition of color data and its mapping to black and white printed output. For example, all pages of a single PDF file are color and some of them are to be printed in color according to some color printing condition, and others to be printed black and white according to some black and white printing condition where a minimum of two OutputIntent entries is required. It was pointed out that a work around exist for this case where the pages to be printed black and white could be prepared as grayscale and in gamut of the black component of the targeted color printing condition.

The further requirement was then raised that the pages of the PDF file targeting a grayscale printing condition must target a black (key) channel only printing process. It was noted that this is desirable because black only printing is less expensive (fewer clicks) than four-color overlay printing. It was pointed out that this relates to how a device produces the intended output color rather than how the color is specified in the file and that this can be addressed in JDF.

It was agreed that the best solution would be to extend PDF and enable Page dictionaries to reference one of several OutputIntent dictionaries present in the PDF file. However, since this goes beyond

PDF/X-4 and PDF/X-5 which support only one OutputIntent "slot", it was suggested that this version of ISO 16612-2 should continue to require a single output intent, knowing that there are ways to work around this "deficiency" for now. However, the committee should begin to put together proposals to submit to TC 171 for consideration for the next version of ISO 32000. Then, these issues can be addressed in future versions of PDF/X and PDF/VT.

**Action Item 08-05:** Rosenthol and Drümmer will draft a proposed request relating to the handling of multiple output intents to be submitted to TC 171 WG 7 for consideration for the next version of ISO 32000.

## 6. Review of comments received on ISO 16612-2, WD6

No specific comments other than those of the Japanese delegation were received. The Japanese comments were reviewed earlier in this meeting (see above).

## 7. Review of WD6 of ISO 16612-2 (N 063)

The following issues relating to PXM were discussed:

- Should there be an annex to standard key value pairs? (Yes)
- 6.5.2, Table 6: Decided that the value shall always be a text string (not integer or float). The Key name (not the value) shall comply with the *XML Specification*, but the value can be any Unicode value. An informative note was added to explain why only a text string is allowed (for compatibility with Xpath matching). It was agreed that the Key name shall follow the rules for XML attribute names after the expansion of # escapes and shall not contain a colon. It was agreed to refer to the PDF Name Registry defined in Annex E of PDF 1.6.
- 6.5.2, Table 6: Added a new key called */V*, a text string, which shall be present and shall be interpreted as the value of the implied **NodeName** of the **VTStruct** within which the PXM dictionary is defined.
- Added a list of PMX Keys and values defined by this part of ISO 16612. It was agreed that if another standard that contains appropriate definitions for Key / values could be identified, this document could reference that standard. One possibility is Jiffi, which was a specification developed by GCA several years ago. We will look into the status of that document.

Additional changes were made to Clause 5 and to the Scope, as a result of discussion. These changes will be reflected in draft #7 of the document.

## 8. Review of JDF 1.4 examples demonstrating linkage to ISO 16612-2 conforming files

This issue was addressed in earlier discussion.

## **9. Review of action items**

A summary of the action items from this meeting is N 073.

## **10. Requirements for next meeting**

Prior to the next meeting in September, additional conference call meetings will be held as in the past, with the first two calls tentatively scheduled for Monday, April 28 and Monday, May 19.

The next meeting will be held in conjunction with the other meetings of TC 130 Working Groups and Plenary, scheduled for 22-27 September in The Netherlands. The committee identified a need for 2 days, preferably Monday and Tuesday of that week.

## **11. Adjournment**

The committee thanked SICOGIF and Ms. Pascale Ginguene for their support of the meeting.