

Tully Talk
November 2001

A look under the hood of a PDF/X-1 file

Friends,

Many of you have e-mailed or called to ask some great questions about PDF/X-1:2001. The sheer number and types of questions clearly indicates that many of you would like to know the facts about PDF/X. I think that's great, and I believe it is now appropriate to delve into the details.

In the previous article "Plain talk about PDF/X-1" I carefully avoided the technical details that make up the PDF/X-1:2001 standard [ISO 15930-1 2001]. The purpose of "Plain talk about PDF/X-1" was to bring to light the challenges we face when engaging in the exchange of digital data. I also wanted to make clear that PDF/X-1 is *not* a new or experimental file format; for PDF/X-1 is PDF. "Plain talk about PDF/X-1" also introduced to most readers some new terms, such as **CGATS** [Committee for Graphic Arts Technologies Standards] and **ISO** [International Standards Organization]. These subjects provide a foundation of PDF/X-1 understanding, and are a prerequisite to learning and knowing more.

Defining terms and relationships

In order to clarify the statement "PDF/X-1 is PDF", allow me to characterize PDF and PDF/X-1.

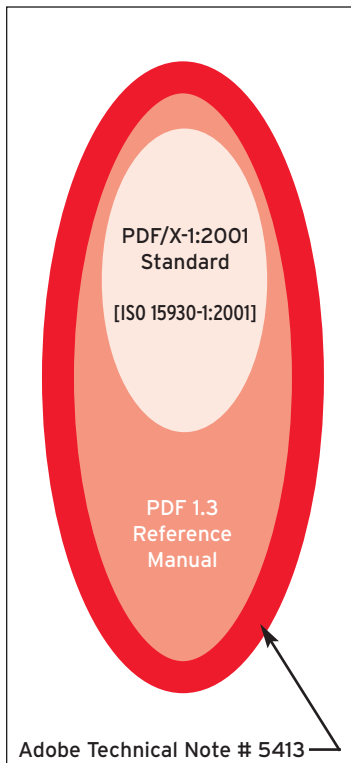
For the present and foreseeable future, the term PDF refers specifically to PDF version 1.3 [Acrobat 4.0x compatibility]. The term PDF is also used to describe a computer *file format*. All that is PDF is exclusively expressed in *Portable Document Format Reference Manual, Version 1.3, Second Edition*.

The term PDF/X-1 specifically refers to *ISO 15930-1:2001*, the PDF/X-1 standard.

The PDF/X-1 standard is a *ruleset* based upon *Portable Document Format Reference Manual, Version 1.3, Second Edition*, as extended by *Adobe Technical Note #5413*. Adobe Technical Note #5413 is an amendment to *Portable Document Format Reference Manual, Version 1.3, Second Edition*, defining additional PDF features and capabilities required for PDF/X-1.

The PDF/X-1 standard defines a nearly static set of conditions to govern the creation, viewing and imaging of specialized PDF files. This specialization is achieved by restricting, requiring or prohibiting specific PDF features and capabilities. Only PDF files meeting the specific conditions of *required*, *restricted*, or *prohibited* can be validated as conforming PDF/X-1 files.

As shown in the diagram [left], you can see that the PDF/X-1:2001 standard is encompassed by both the PDF Reference Manual and Adobe Technical Note #5413. The relationship between the PDF specification and the PDF/X-1 standard is one of inclusion; where PDF/X-1 is a *subset* of the PDF 1.3 specification. As you can see, a PDF/X-1 file is a PDF file.



- = PDF v. 1.3 Specification
- = Technical Note #5413
- = PDF/X-1:2001 Standard

Scott B. Tully
PDF Workflow Evangelist

12 Corporate Drive
North Haven, Connecticut 06473
T 203.985.3718
M 860.388.7572
E stully@aps.vertisinc.com
www.vertisinc.com



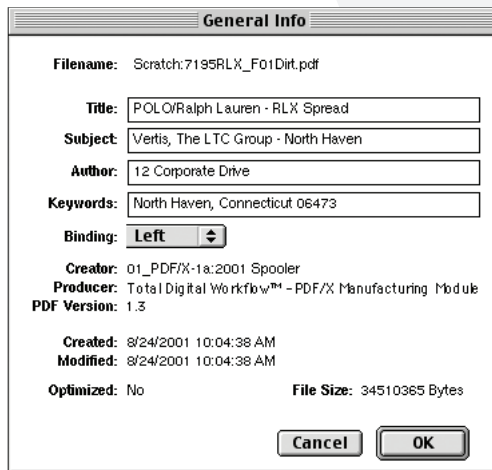
What is so special about a PDF/X-1 file?

Structure and content.

PDF/X-1 files are created with a single purpose in mind– to facilitate the “blind exchange” of graphically rich content ready for imaging. To accomplish this, it is necessary to limit the content and structure of the PDF/X-1 file to ensure it’s imaging integrity. The limiting of content is divided into four (4) conditions:

Required	A conforming PDF/X-1 file shall contain this object or key.
Prohibited	A conforming PDF/X-1 file shall not contain this object or key.
Restricted	Certain values or combinations of objects and keys where the contents are required or prohibited.
Recommended	It is recommended that all conforming files contain this key.

For example, a PDF file can contain movies, sounds, and file attachments. These valid PDF features called *annotations*, are unsuitable for the digital manufacturing processes, and are *prohibited* in a PDF/X-1 file. Other annotations such as notes, stamps, and Acrobat Forms are PDF features that may add value to a PDF/X-1 file. Their presence in a conforming PDF/X-1 file is *restricted*, for they must be positioned outside the bleed area *and* must not obscure any portion of the imageable area when displayed on screen.



A majority of the *required* and/or *recommended* conditions specified within the PDF/X-1 standard are expressed as *arrays* and *key-value* pairs. These PDF programming elements are contained within *dictionaries*.

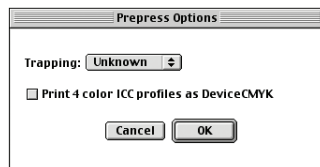
A familiar, visual example of a dictionary is the **Info** dictionary expressed as **General Info** [left].

The **Info** dictionary in every PDF contains default key-value pairs of **Title**, **Subject**, **Author**, **Keyword**, **Creator**, **Producer**, **CreationDate**, **ModDate**, and **Trapped** as defined in the *Portable Document Reference Manual, Version 1.3, Second Edition*. Of these nine (9) default keys, six (6) are *required* by the PDF/X-1 standard, two (2) keys: **Creator** and **Producer** are *recommended*, and the **Trapped** key is *restricted*.

The PDF/X-1 standard introduces two (2) additional *required* **Info** dictionary key-value pairs: **GTS_PDFXVersion** and **GTS_PDFXConformance**. These keys are unique to PDF/X files, and they are not named in the *Portable Document Reference Manual, Version 1.3, Second Edition*. Since these keys are not defaults, they must be generated by a compliant PDF/X-1 workflow or tool.

Prepress particulars

The use of the **Trapped** key is of particular interest to PDF/X-1 creators. While it is a default key-value pair within the **Info** dictionary, the condition of the **Trapped** key is not presented in the **General Info** dialog box.



In order to view the condition of the **Trapped** key, the **Prepress** tool must be invoked [File->Document Info->Prepress...]. Invoking the **Prepress** tool produces the **Prepress Options** dialog box, where the condition of the **Trapped** is key presented in the form of a pull-down menu [left]. This is an incredibly misleading display, for the available choices of **Yes**, **No** or **Unknown** imply action upon selection. This is simply not the case, for the **Prepress** tool does not perform any PDF trapping, it simply allows users to interactively set the condition of the **Trapped** key.

The default condition of the **Trapped** key in every PDF file is **Unknown**, and this condition is *prohibited* by the PDF/X-1 standard.

The reasoning behind this is clear— a “blind exchange” cannot occur where the file being exchanged contains any “unknown” parameters. A PDF/X-1 creator [or PDF/X-1 compliant tool or workflow] is obligated to properly set the **Trapped** key in a PDF/X-1 file so that the receiver is informed, and to enable a PDF/X-1 compliant workflow or tool to automatically respond.

Therefore, the condition of the **Trapped** key is *restricted* in a PDF/X-1 file, requiring the creator of a PDF/X-1 file to select either: **True** or **False** [represented by **Yes** or **No** the **Prepress Info** dialog box]. Files that have been completely trapped must have the **Trapped** key set to **True**. Files containing no traps must have the **Trapped** key set to **False**. Partially trapped files are not permitted.

Arrays

Like key-value pairs, an *array* is a common building block of PDF. An array is used to define *literal names*, *integers*, and *strings*. Arrays are often used to associate multiple values for a single *variable*.

While PDF files contain many arrays, PDF/X-1 creators must be familiar with (4) four in particular: the **MediaBox**, **BleedBox**, **TrimBox**, and **ArtBox**. As their names suggest, each array defines a rectangular area of a PDF file. The inclusion/exclusion of these arrays in a PDF/X-1 file is carefully limited, requiring a PDF/X-1 creator or workflow tool to properly “mark” a PDF/X-1 file with the appropriate combination of arrays.

By default or *inheritance*, every PDF file contains a **MediaBox**. The **MediaBox** represents the physical size of a PDF page, and is derived during the PDF creation process from either the **PageSize** array [in the source PostScript file], the **BoundingBox** array [in an encapsulated PostScript file] or from a default value determined by the *normalizer* [application used to create a PDF file]. Of the (4) four arrays, only the **MediaBox** is both a PDF default, and a *required* array in a PDF/X-1 file.

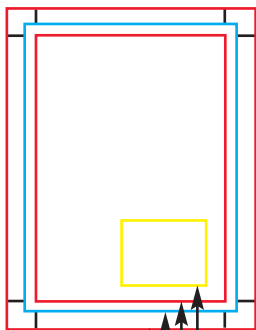
The creation and placement of the remaining (3) three arrays is essential to PDF/X-1 conformance.

A conforming PDF/X-1 file must contain either the **TrimBox**, or **ArtBox** but not both. Since the usage of these two arrays is conditional, they are, in PDF/X terms, *restricted*. To determine which array is appropriate for use, the function and intent of each array must first be defined.

The **TrimBox** is used to define the area of a PDF file which represents the final size of the page after printing and finishing. Customarily, this page area is indicated by the use of *crop marks*. A **TrimBox** should only be used when the PDF/X-1 file is a complete page. The inclusion of a **TrimBox** in a PDF/X-1 is intended to facilitate “blind placement” of a PDF/X-1 file in an electronic *imposition*. If the **TrimBox** array is present, its boundaries must not exceed those of the **MediaBox**.

In the case where the PDF/X-1 file is not a complete page, or only a portion of the page is to be used for display or imaging, the **ArtBox** is to be used to indicate the desired image area. The function of the **ArtBox** is to act as an electronic “crop” without physically changing the dimension of the PDF/X-1 file. The inclusion of an **ArtBox** serves to indicate the imaging intent of the PDF/X-1 files creator, and this intent is to be honored by the receiver of a PDF/X-1 file. Like the **TrimBox**, the boundaries of an **ArtBox** must not exceed those of the **MediaBox**.

The remaining array, the **BleedBox**, is used to define the imaging area in excess of the trim. The use of the **BleedBox** is *optional*, and its inclusion or exclusion is left to the discretion of the PDF/X-1 files creator. When the **BleedBox** is included, its boundaries must reside within those of the **MediaBox** and may not encroach upon the **TrimBox** or **ArtBox**.



MediaBox
BleedBox
TrimBox
ArtBox

Fonts, Compression and Colorspace

A valid PDF/X-1 file must always have all fonts embedded, and a receiver must always use the embedded fonts for display and imaging. Fonts used in a PDF/X-1 file must always be legally embeddable, thus honoring existing copyrights of some font owners. A creator of a PDF/X-1 file is expected to ensure that all fonts used in a PDF/X-1 file are used in compliance with licensing agreements.

Compression of **Image XObjects** [bitmapped images] contained with a PDF/X-1 file is not required. If compression is to be employed in the creation of a PDF/X-1 file only PDF-supported *Flate*, *RunLength*, *JPEG* or *CCITT Fax* compression may be used. The effective use of the appropriate compression algorithm and settings is left to the discretion of the PDF/X-1 files creator.

The proper use of colorspace is of great importance to both the PDF/X-1 creator and receiver. The content of a PDF/X-1 file may be defined in **DeviceCMYK**, **DeviceGray**, **Separation**, **DeviceN**, **Indexed**, or **Pattern** colorspace. The use of each of these different colorspace in a PDF/X-1 file is *restricted*. Non-printing objects may employ any PDF supported colorspace. Spot colors are permitted in a PDF/X-1 file and must be defined in either the **Separation** or **DeviceN** colorspace. The use of the following *device dependent* and *device independent* colorspace: **DeviceRGB**, **DefaultRGB**, **CalGray**, **CalRGB**, and **Lab** are *prohibited*.

When objects are marked in either the **Separation** or **DeviceN** colorspace, they must have their respective *alternateSpace* resources defined as either **DeviceCMYK** or **DeviceGray**. Objects marked in either **Indexed** or **Pattern** must have their *base* colorspace defined as **DeviceCMYK**, **DeviceGray**, **Separation** or **DeviceN** as per the previous restrictions.

Objects with a single tonal range require special attention. Spot colors must be defined as either **Separation** or **DeviceN** colorspace, and objects marked as “Black” may employ either the **DeviceGray**, or **Black Separation** colorspace. The effective use of the proper colorspace has a significant effect when marking “Black” objects, for the *overprinting* behavior of the **Separation** colorspace differs from the **DeviceGray** colorspace. To effectively overprint “Black” objects marked in the **Separation** colorspace, the **OPM** key in the *extended graphics state* of that object must be set to 1.

The restricted use of colorspace in a PDF/X-1 file is necessary to facilitate the functionality of the **OutputIntents** array contained in the **Catalog**. This unique PDF/X-1 array is not a part of the *Portable Document Format Reference Manual, Version 1.3, Second Edition*. The **OutputIntents** array, **OutputIntents** dictionary and **OutputConditionIdentifier** key are defined in *Adobe Technical Note #5413*; a requirement of the PDF/X-1 standard. The purpose of this requirement is to communicate the color rendering intent of the creator to the receiver. The **OutputIntents** array contained in the **Catalog** object is a vehicle to technically identify the *characterized printing condition* [ie. Offset, Gravure, Flexographic] of the data contained in the PDF/X-1 file by employing existing color management standards established by the ICC [International Color Consortium].

In order to communicate the *characterized printing condition* of a PDF/X-1 file, the **OutputIntents** array must contain a single **OutputIntents** dictionary whose **S** key value must be expressed **/GTS_PDFX**. This key is called the **OutputIntent** object. The **OutputIntent** object must also have a corresponding **OutputConditionIdentifier** key. Acceptable values of this key can be found in the ICC registry of characterized printing conditions. When the value of **OutputConditionIdentifier** key matches a characterization found in the ICC registry, the **RegistryName** key must be present with the value (<http://www.color.org>).

Conversely, if the PDF/X-1 files creator employs a key not found in the ICC registry of characterized printing conditions, a **DestOutputProfile** key must be present in the **OutputIntent** object, and the **OutputCondition** key should be present. If the **OutputCondition** key is present, a string value describing the intended printing condition should be used. This string value should be expressed in plain language, and be meaningful to a PDF/X-1 receiver.

An example of this string could be: (This file was prepared for SWOP v. 2.0).

Admittedly, the technical information provided to properly cover both the colorspace and characterized printing condition requirements of PDF/X-1 is mind-numbing to all but a PDF/X-1 developer or programmer. It has been included in this document to demonstrate the lengths to which an implementer of a PDF/X-1 workflow must go to ensure compliance. Keep in mind that full comprehension of the material presented thus far *is not* a prerequisite for success, for the mechanisms needed to fulfill the characterized printing condition requirement can be found in PDF/X-1 tools today.

Alternate Images

The *Portable Document Format Reference Manual, Version 1.3, Second Edition* provides for the inclusion of *alternate images* in a PDF file. An alternate image is secondary copy derived from a source **Image XObject** and is stored in the **Alternate** array. The alternate image can be an exact duplicate of the parent **Image XObject**, where its colorspace, compression and *bit depth* are preserved *or* the alternate image could be a highly compressed low resolution *replica*. An effective use of this PDF native feature would be to create a low resolution [72 dpi], JPEG compressed alternate image for each original high resolution image. When a file containing alternate images is displayed on screen [with an alternate image enhanced viewer] the rate at which the display can render the alternate images is exponentially faster than the rate at which it would render the high resolution original. This display efficiency lends itself to expedited approval procedures, both local and remote.

The PDF/X-1 standard recognizes both this capability and opportunity by allowing the *restricted* use of alternate images. The PDF/X-1 standard stipulates that when alternate images are present in the **Alternate** array, those alternates must have their **DefaultForPrinting** key set to **False**, and that alternate images shall represent the same area in the display as would its high resolution original. It also stipulates that if/when a high resolution original is edited, the alternate image must be immediately regenerated or be destroyed. The restrictions placed on alternate images in PDF/X-1 ensures that relationship of high resolution original(s) [parent] and its alternate image(s) [children] are always synchronous.

Conformance levels– PDF/X-1 and PDF/X-1a

The *Portable Document Format Reference Manual, Version 1.3, Second Edition* provides for document security by including an *encryption* feature. Known as **Standard** encryption, this Adobe Acrobat exclusive means of securing, or limiting access rights to a PDF file is *restricted* by the PDF/X-1 standard. **Standard** encryption may be employed by the PDF/X-1 files creator provided that the access rights to viewing and **Printing** are never denied, and that when one or more of the remaining **Standard** encryption features are employed, the password must be an empty string. The status of the **Standard** encryption features are contained in the **Encrypt** dictionary in the form of *key-value* pairs.

Within the PDF/X standards community, there were those who objected to the application of document security. These persons strongly believed that restricted access of any kind violated the “spirit” of “blind exchange”, and successfully argued for an exception; a second *conformance level* of the PDF/X-1 standard. This second conformance level is expressed as PDF/X-1a:2001, where the use of **Standard** encryption is *prohibited*.

As the PDF/X-1:2001 standard was being upgraded from the previous PDF version 1.2 based ANSI [American National Standards Institute] CGATS 12/1:1999 edition, a second contested issue arose. The use of OPI [Open Prepress Interface] within a “blind exchange” was strongly debated by two opposing points of view. One camp argued that OPI references to embedded files, which was allowed in CGATS 12/1:1999 [the original PDF/X-1 standard], was still needed to facilitate the inclusion of legacy file formats such as *DCS 2.0, [Desktop Color Separation, Version 2.0] Copydot scans* [TIFF 6.0 or EPS] and *TIFF/IT-PI* [ISO 12639 *Graphic Technology– Prepress digital data exchange- Tag image file format for image technology*] in a PDF/X-1 file.

The opposing camp argued that such legacy file formats had no place in a PDF file, and if they were to be used in an exchange, they more suitably exchanged in their native form.

Compromise prevailed, resulting in the *restricted* use of OPI externally referenced files PDF/X-1, and the absolute *prohibition* of OPI in PDF/X-1a.

The easiest approach to understanding and remembering the different *conformance level* requirements of the PDF/X-1:2001 standard, is to think of PDF/X-1 as “the rules” and PDF/X-1a as the rules– with two exceptions: the prohibited use of encryption and OPI.

Commentary

The technical information presented in this document represents a significant portion of the PDF/X-1 standard. It is not, and was never intended to be, a complete account of the entire standard. You can secure a printed copy of the PDF/X-1 Standard and Application Notes by visiting the NPES website: <http://www.npes.org/standards/workroom.html>.

I encourage you to contact me personally [stully@aps.vertisinc.com] with any PDF, PDF/X and PDF for CTP related questions you may have, or to offer suggestions for future editions of *Tully Talk*.

Mr. Scott Tully is the PDF Workflow Evangelist for Vertis Inc. Scott specializes in designing and implementing automated PDF workflows for Prepress, Computer to Plate and e-commerce via the Internet. A "do-it-yourself" PDF pioneer, and champion of PDF/X Scott is well known for successfully engineering a "homegrown" fully automated, color-managed, trapped PDF workflow that delivers PDF/X-1a:2001 files and digital color contract proofs.



Scott is a Adobe Acrobat 5.0 Certified Expert, a member of the Adobe Solutions Network Advisory Board, International Prepress Association (IPA) Technical Standards Committee, Digital Distribution of Advertising for Publications Association (DDAP) and the Committee for Graphic Arts Technical Standards (CGATS) Task Force 1 Workgroup 6.