PDFTron XPSConvert™
User Manual

Version 1.x

LEGAL STATEMENT AND COPYRIGHT NOTICE

PDFTron XPSConvert™ Command-Line Application User Manual
Part number: PDFTRON-1-XPSConvertCMD
Last Updated: February 1, 2010

TABLE OF CONTENTS

1. Introduction

1.1 An Introduction to PDFTron XPSConvert

PDFTron's XPSConvert is an easy-to-use, multi-platform command-line program that provides users with an efficient way to convert XPS or OpenXPS documents into PDF as well as image formats (such as PNG, TIFF, BMP, and JPEG).

XPSConvert enables high-quality conversion from XPS to PDF that maintains the original document quality and preserves hyperlinks, colors and fonts. The resulting self-contained and compact PDF and image files can be distributed, viewed, edited, archived, printed, and published. The conversion also offers a wide range of options to control the output file size and image quality.

Like other PDFTron products, XPSConvert does not rely on Microsoft .NET platform or any other third party components. The technology is suitable for use in high-throughput server environments and is also available as a component for integration with third party applications.

1.1.1 Key Functions

- Convert XPS or OpenXPS to PDF
- Convert XPS or OpenXPS to PNG, PNG8, JPEG, TIFF, TIFF8, TIFF (G3/G4 CCITT Fax) BMP, and RAW.
- Full support for XPS and OpenXPS specifications.
- Batch conversion.
- Conversion support for unzipped XPS folders.
- Preserves embedded fonts, ICC profiles, and hyperlinks.
- Multi-page Tiff export.
- Dithering control for 1 bit per pixel and palletized image formats.
- Wild card and subfolder processing.
- Adjustable output resolutions for raster image output.

1.1.2 Common Use Case Scenarios

- Developers may want to use XPSConvert to quickly add XPS support to any applications or workflows that support PDF.
- Server-based, on-demand conversion of XPS documents to PDF files or raster images.
- Batch processing of PDF collections with same rasterization options. XPSConvert is particularly useful in assembling product catalogues and brochures.
- Thumbnail generation for XPS documents.

1.1.3 Operating Systems Supported

- Windows 7, 2008, Vista, XP, 2003, 2000, NT, 98
- Mac OSX
- Linux

1.1.4 System Requirements

- At least 10 MB of free disk space.
- Memory requirement is dependent on source document being converted.

1.2 XPSConvert SDK

For developers who are looking for a software development component to integrate into their application, PDFTron also offers **XPSConvert SDK**, an easy-to-use, yet powerful software component for embedding into client and server based applications. XPSConvert SDK is available as a shared library and can be easily accessed from any programming language (including C#, VB.NET, C/C++, Java, VB6, Perl, Python, Ruby, Delphi, etc). XPSConvert SDK is also available as an add-on module for PDFNet SDK.

1.3 About This Manual

This manual is intended as a guide to the installation and use of XPSConvert. It is intended for programmers and other users who are familiar with XPS documents, graphic image file creation, graphic file manipulation and general computer processes.

- Section 1 introduces XPSConvert and describes the manual.
- Section 2 explains how to install and uninstall XPSConvert.
- Section 3 covers basic use of XPSConvert.
- Section 4 is where you will find all the support information you may require, such as how to report a problem with the software.

2. Installing and Uninstalling XPSConvert

2.1 XPSConvert Installation

XPSConvert Command-line Application is supplied as a download from a distributor or directly from www.pdftron.com. The release is packaged as a .zip file (XPSConvert.zip). To install the software, simply unzip the archive in the desired location and make sure to preserve the directory/folder structure during this process. To register the software, copy the license file provided to you into the "XPSConvert" folder.

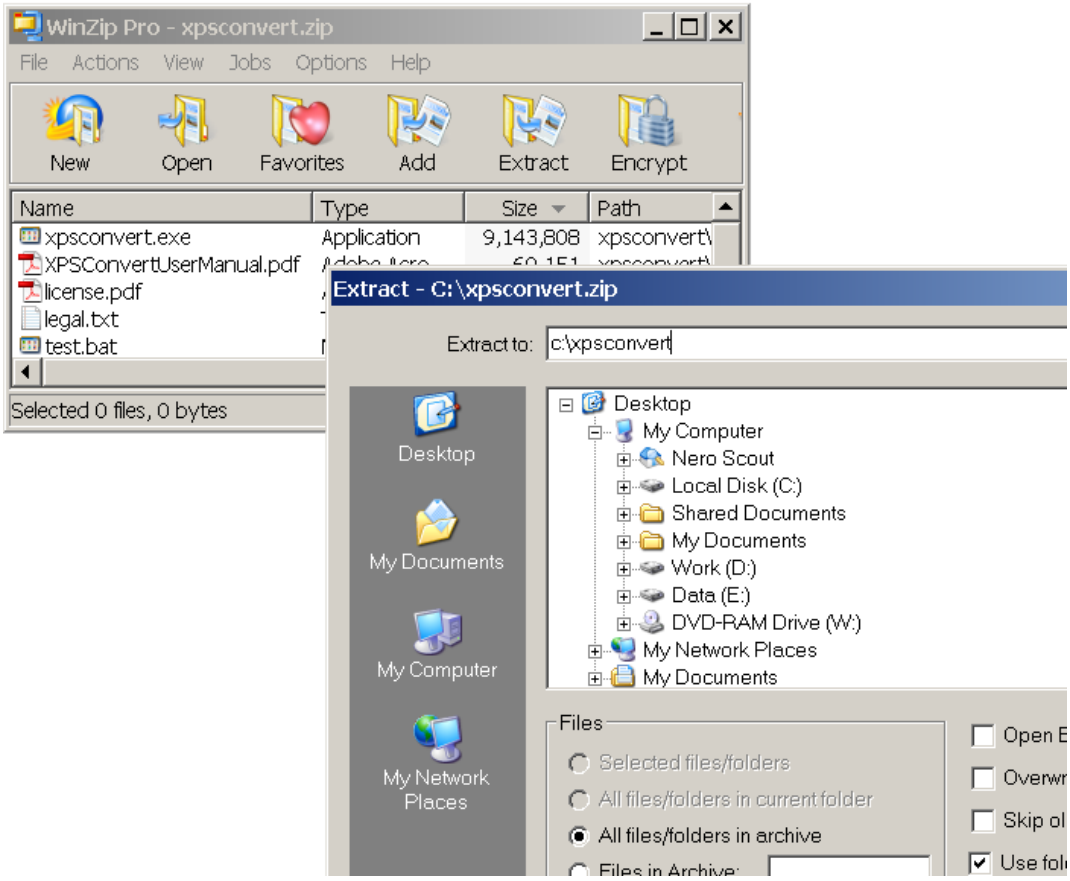

Figure 2.1 – Extracting XPSConvert Archive using WinZip

2.2 Demo Version Installation

If you wish to evaluate the product, you can download the demo version of the product without any serial number or license key.

To do this, go to PDFTron's **Downloads** page at www.pdftron.com/downloads.html. Click on the appropriate product name/version. This will bring you to the link to the page to download the demo. Download the zip file (XPSConvert.zip) and extract the archive in the desired location, while making sure to preserve the directory (folder) structure when extracting the archive. Download the zip file *xpsconvert.zip*. Extract the archive in the desired location (making sure to preserve the folder structure). This will provide you a working copy of the application along with various examples. The limitation of the evaluation version is that all output pages will have demo stamp.

2.3 Uninstalling XPSConvert

To remove XpsConvert from your computer, simply delete the “xpsconvert” folder.

3. Overview

PDFTron XpsConvert is a command-line application designed to convert XPS documents or XPS uncompressed folders to one or more PDF, BMP, PNG, JPEG, TIFF, or RAW files while presenting several options to control resolution, color, and quality depending on the output format selected. This section covers the basic usage of XPSConvert explaining all of the available options.

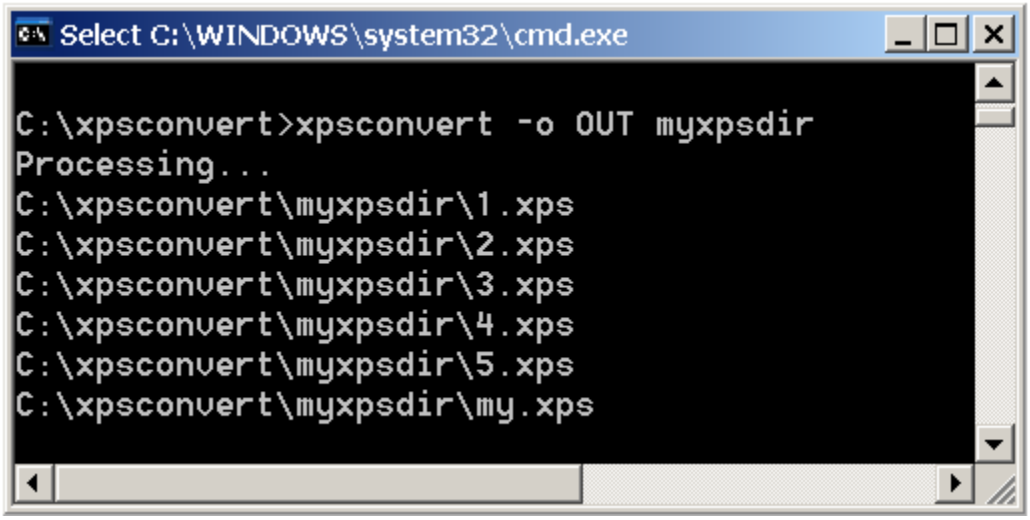

Figure 3.0 XPSConvert Command-line Application.

3.1 Basic Syntax

The basic command-line syntax is:

```
XPSConvert [options] file1 file2 folder1 file3 ...
```

3.2 Command-Line Summary

The following is a list of available command-line options for XPSConvert:

Option	Parameter	Description
-o or --output	-o myfolder -o c:\myfolder	The output folder used to store converted files. The default output folder is the current working folder.
--subfolders		Process all sub-directories for every directory specified in the argument list. By default, sub-directories are not processed.
--prefix	--prefix myprefix	The prefix for the output file. The output filename will be constructed by concatenating the prefix string, the page number (when converting multi-page documents to image) and the appropriate extension (e.g. myprefix.pdf or myprefix_2.png). The prefix option should be used only for conversion of individual documents. By default, each input filename will be used as a prefix.
-f or --format	-f pdf -f jpg -f bmp -f tif -f png	Output format. The following is a list of supported export formats: ▪ **pdf** (Portable Document Format; ISO32000) ▪ **png** (Portable Network Graphics) ▪ **png8** (Palletized PNG) ▪ **jpg or jpeg** (Joint Photographic Expert Group) ▪ **tif** or **tiff** (Tagged Image File Format) ▪ **tif8** (Palletized Tagged Image File Format) ▪ **bmp** (Windows Bitmap Format) ▪ **raw** (raw RGB data) The default output format is **pdf**.
-a or --pages	Convert page 1,3, and 10: -a 1,3,10 Convert all even pages: -a even Convert pages in the range from 3-11 and page 50: --pages 3-11,50 Convert all odd pages and all pages in the range from 100 to the last page: -a odd,100-	Specifies the list of pages to convert. By default, all pages are converted.
-r or --rotate	-r 90 -r 180 -r 270	Rotates all pages by a given number of degrees, counterclockwise. The allowed values are: ▪ **0** ▪ **90** ▪ **180** ▪ **270** The default value is 0.

--physical_model		Specifies the physical model of the xps files to process. The following is a list of supported physical models: ▪ **zip (the standard format of xps files)** ▪ **dir (directories packaged like xps files)** ▪ **all (both zip files and directories)** The default physical model is **zip**.
--nosmooth		Disables image smoothing.
--extension	--extension ".xps"	The default file extension used to process PDF documents. The default is ".xps".
-h or --help		Print a listing of available options.
-v or --version		Print the version information.
--verb	--verb 2	Set the verbosity level. Valid parameter values are 0, 1, and 2. The higher number results in more feedback. The default is 1.

The following command line options apply only to image conversion:

Option	Parameter	Description
--digits	--digits 4	The number of digits used in the page counter portion of the output filename. By default, new digits are added as needed; however this parameter could be used to format the page counter field to a uniform width (e.g. myfile_0001.png, myfile_0002.png, etc).
-d or --dpi	-d 300	The output resolution, from 1 to 1000, in Dots Per Inch (DPI). The higher the DPI, the larger the image. Resolutions larger than 1000 DPI can be achieved by rendering image in tiles or stripes. The default resolution is 92 DPI.
--hres	--hres 100	The width of the output image, in pixels.
--vres	--vres 100	The height of the output image, in pixels.
-c or --clip	-c 216,522,330,600	User definable clip box. By default, the clip region is identical to the current page 'box'.
-g or --gray	--gray	Render and export the image in grayscale mode. Sets pixel format to 8 bits per pixel grayscale. By default, the image is rendered and exported in RGB color space.
-k or --cmyk	--cmyk -f tif	Render and export the image in CMYK mode. To export CMYK, the output image format must support CMYK pixel format. An example of image format that supports CMYK is TIFF (e.g. -f tif -k). By default, the image is rendered and exported in RGB color space.
--mono	--mono	Export the rendered image as 1 bit per pixel (monochrome) image. If the output format is TIFF, the image will be compressed using G4 CCITT compression algorithm. By default, the image is not dithered. To enable dithering use '--dither' option.
--dither	--dither	Enables dithering when the image is exported in palletized or monochrome mode (e.g. when export format is tif8, png8 or --mono).

--gamma	--gamma 0.3	Sets the gamma factor used for anti-aliased rendering. Typical values are in the range from 0.1 to 3. Gamma correction can be used to improve the quality of anti-aliased image output and can (to some extent) decrease the appearance of common anti-aliasing artifacts (such as pixel width lines between polygons).
-q or --quality	-q 100	Compression quality is a number in the range from 1 to 100. Lower numbers usually result in better compression at the expense of image quality. The default setting is 80.
-m or --multipage	--multipage	If the output image format supports multi-page or multi-frame capability, store all output images in one file instead of separate files. Currently, this option is only relevant to TIFF output. By default, images will be saved in separate files.

3.3 Basic Usage

3.3.1 How do I save converted files in a given folder?

By default, XPSConvert saves converted files in the current working folder. To specify another output location, use the '-o' (or --output) parameter. For example:

```
xpsconvert –o "c:\My Output" 1.xps 2.xps 3.xps
```

Note: If the specified path does not exist, XPSConvert will attempt to create the necessary folders.

3.3.2 How can I control the output name for converted files?

When converting to PDF, XPSConvert will, by default, create a single file with the name of the input XPS file. The output filename can be changed using the '--prefix' option. For example, the following command-line generates an output document named outdoc.pdf:

```
xpsconvert –prefix outdoc mydoc.xps
```

The default behavior for image conversion is to create a separate image file for every page in the document. The output filename is constructed using the name of the input XPS file, page counter, and appropriate image extension. For example, the following command-line generates a sequence of image files starting with mydoc_1.jpg, mydoc_2.jpg, etc.:

```
xpsconvert –f jpg mydoc.xps
```

XPSConvert allows output filename customizations using the '--prefix' and '--digits' options. For example, the following command-line generates a sequence of image files starting with newname_0001.jpg, newname_0002.jpg, etc.:

```
xpsconvert –f jpg --prefix newname --digits 4 mydoc.xps
```

The '--digits' parameter specifies the number of digits used in the page counter portion of the output filename. By default, new digits are added as needed, however this parameter could be used to format the page counter field to a uniform width (e.g. myfile_0001.jpg, myfile_0010.jpg, instead of myfile_1.jpg, myfile_10.jpg, etc).

If your output image format is TIFF, you can convert XPS to a single, multi-page TIFF document using the '--multipage' option (See 'How do I convert XPS to multi-page TIF?' for an example).

To avoid any ambiguities in file naming, the prefix option should be used only for conversion of individual documents.

3.3.3 How do I convert XPS to PDF, JPEG, PNG, TIF or some other format?

By default, XPSConvert automatically converts XPS to PDF. The output format can be modified using the '-f' (or --format) option. For example,

```
xpsconvert –f jpg in.xps
```

will convert XPS to JPEG.

The '--format' parameter accepts any of the following output formats:

- **pdf** – (Portable Document Format)
- **png** - (Portable Network Graphics)

- **png8** – (Palletized PNG)
- **jpg or jpeg** (Joint Photographic Expert Group)
- **tif** or **tiff** (Tagged Image File Format)
- **tif8** – (Palletized TIFF)
- **bmp** (Windows Bitmap Format)
- **raw** (raw RGB or Gray data)

3.3.4 How do I specify which pages to convert?

By default, XPSConvert will convert all XPS pages to the output format. You can specify a subset of pages to convert using the '-a' or '--pages' options. For example:

```
xpsconvert -a 1,3,10 in.xps
```

will convert only pages 1, 3, and 10. Please note that XPSConvert assumes that all pages are numbered sequentially starting from page 1.

To specify a range of pages, use dash character between numbers. For example:

```
xpsconvert -a 1,10-20,50- in.xps
```

will render the first page, pages in the range from 10 to 20 and all pages starting with page 50 to the last page in the document.

All even pages can be selected using the 'e' (or 'even') string. For example, the following line renders all even pages:

```
xpsconvert --pages even in.xps
```

Similarly odd pages can be selected using the 'o' (or 'odd') string. The following line renders all odd pages in the document and every page in the range from 100 to the last page:

```
xpsconvert --pages odd,100- in.xps
```

3.3.5 How can I rotate pages?

The resulting rotation can be modified using the '-r' (or --rotate) option. For example, the following line rotates all pages 90 degrees counterclockwise:

```
xpsconvert --rotate 90 Test/tiger.xps
```

Similarly, the following line rotates the page 270 degrees counterclockwise (or 90 degrees clockwise):

```
xpsconvert --rotate 270 Test/tiger.xps
```

3.3.6 How do I convert unzipped XPS files?

By default, XPSConvert will convert XPS files if they are in zip format. You can convert directories in XPS format by using the option --physical-model. For Example:

```
xpsconvert --physical_model dir xps_dir
```

In this example 'xps_dir' could be created by unzipping an XPS file. This directory would contain files with XML markup and other XPS resources.

The "--physical_model" parameter accepts any of the following output formats:

- **zip** (xps parts contained in a zip archive)
- **dir** (xps parts contained in a directory)
- **all** (processes both zip files and xps directories)

3.3.7 How do I batch convert files?

XPSConvert supports batch conversion of many PDF files in a single pass. To convert all PDF files in a given folder(s) you can use the following syntax:

```
xpsconvert myfolder1
```

The '--subfolders' option can be used to recursively process all subfolders. For example, the following line will convert all documents in 'myfolder1' and 'myfolder2' as well as all subfolders:

```
xpsconvert --subfolders myfolder1 myfolder2
```

By default, XPSConvert will convert all files with the extension '.xps'. To select different files based on the extension use the '--extension' parameter. For example, to convert all XPS documents with a custom extension '.blob', you could use the following line:

```
xpsconvert --extension .blob --subfolders myfolder1
```

The use of wild characters is also allowed. For example, to convert all XPS files starting with 'x' in the current folder use:

```
xpsconvert x*.xps
```

3.3.8 How do I convert XPS to multi-page TIF?

If your output format is TIFF, you can convert XPS to a single, multi-page TIFF document instead of a separate file for every page using the '--multipage' option.

For example:

```
xpsconvert --multipage -f tif --verb 3 myxps.xps
```

3.3.9 How do I create grayscale images?

By default, XPSConvert uses the RGB color model for rasterization and image export. You can instruct XPSConvert to use single channel Device Gray color model for rasterization and image export using the '--gray' option. For example:

```
xpsconvert -f tiff --gray in.xps
```

3.3.10 How do I specify the resolution of the output image?

Using XPSConvert output image resolution can be specified explicitly (using the '–d' or '--dpi' option) or implicitly (using the '--hres' and '--vres' parameters). In this section, we cover the use of the '--dpi' parameter. For more information on the '--hres' and '--vres' parameters, see 'How do I specify dimensions of the output image in pixels?'

By default, XPSConvert uses resolution of 92 Dots Per Inch (DPI), which is the typical screen resolution. Smaller DPI numbers result in smaller images (e.g. suitable for use as thumbnails), while larger DPI numbers generate larger images (e.g. suitable for high-quality output).

For example, to convert a XPS document to a multi-page TIF at 300 DPI (Dots Per Inch), use the following line:

```
xpsconvert -f tif --multipage --dpi 300 in.xps
```

Depending on the dimensions of the input page, high DPI/resolution rasterization requires lots of memory. For example, rasterization of a single A4 page (8x11) at 1000 DPI will require more than 350MB of memory. If XPSConvert fails to allocate enough memory, you can render the image in stripes or tiles, as described in 'How do I render high-resolution images', or by trying to decrease DPI value.

A 'typical' range of acceptable DPI values is between 1 and 1000 DPI. XPSConvert can rasterize images beyond 1000 or 2000 DPI using tiled or striped rendering.

3.3.11 How do I specify dimensions of the output image in pixels?

To specify absolute dimensions of the output image in pixels, use the '--hres' and '--vres' parameters.

When these parameters are specified, XPSConvert will automatically determine the DPI (Dots Per Inch) ratio required to match the pixel dimensions of the output image.

For example, to generate 100 by 100 pixels thumbnails for a given XPS, you can use the following line:

```
xpsconvert -f jpg --hres 100 --vres 100 in.xps
```

Because the input XPS page may not perfectly fit the absolute pixel size of the output image, XPSConvert will also center the page and preserve the aspect ratio during rendering.

To generate images that are proportional in their size to the input XPS pages, simply omit one of the parameters (either --hres or --vres). For example,

```
xpsconvert -f jpg --hres 100 in.xps
```

will convert all XPS pages to images that are 100 pixels wide, with height proportional to the dimensions of the input page.

Similarly, the following line will create images with fixed height (100 pixels) and variable width (to preserve the aspect ratio).

```
xpsconvert -f jpg --vres 100 in.xps
```

3.3.12 How do I render only a subset of a given page?

Using XPSConvert you can rasterize a subset of a page using the '--clip' parameter. The parameter accepts a list of four numbers, separated using commas, giving the coordinates of a pair of diagonally opposite corners. Typically, the list takes the form: *llx, lly, urx, ury* specifying the lower-left *x*, lower-left *y*, upper-right *x*, and upper-right *y* coordinates of the rectangle, in that order. The other two corners of the rectangle are then assumed to have coordinates (*llx*, *ury*) and (*urx*, *lly*). All coordinates need to be expressed in points (a basic unit of PDF 'user' coordinate system). One PDF point is 1⁄72 of an inch and is approximately the same as a point (unit commonly used in the printing industry).

The '--clip' parameter is not only useful for cropping pages, but it can be also used to speed up the rendering process and to reduce memory consumption (see 'How to I render very large images?' for details).

XPSConvert also supports clipping to predefined page regions, such as page media, crop, trim, bleed, and art box. For more information on clipping to predefined regions, see 'How can I show/hide crop marks or the trim region?

3.3.13 How do I render very large images?

Depending on the dimensions of the input page, high DPI/resolution rasterization requires lots of memory. For example, rasterization of a single A4 page (8x11) at 1000 DPI will require more than 350MB of memory. If XPSConvert fails to allocate enough memory (a single contiguous block of memory), you can render the image in stripes or tiles by repeatedly rendering different regions of the page using the '--clip' parameter (also see 'How do I render only a subset of a given page?').

For example, if the input page has a media box 0,0,595,842, you could render the page at 2000 DPI (Dots Per Inch) in four stripes (using 210.5 point increments along the Y axis) as follows:

```
xpsconvert --dpi 2000 --clip 0,0,595,210.5   --prefix t01 Test/tiger.xps
xpsconvert --dpi 2000 --clip 0,210.5,595,421 --prefix t02 Test/tiger.xps
xpsconvert --dpi 2000 --clip 0,421,595,631.5 --prefix t03 Test/tiger.xps
xpsconvert --dpi 2000 --clip 0,631.5,595,842 --prefix t04 Test/tiger.xps
```

Rendering of the same image in a single pass would require more than 1.4 GB in memory.

3.3.14 How do I specify compression ratio for JPEG format?

The JPEG image format offers a lossy type of compression and the option to trade between the loss in image quality and compression ratio. To fine-tune JPEG compression quality, use the '--quality' parameter as illustrated in the following sample:

```
xpsconvert --quality 80 -f jpg Test/tiger.xps
```

Compression quality is a number in the range from 1 to 100. Lower numbers usually result in better compression at the expense of image quality. The default is 80.

3.3.15 How do I render XPS as CCITT Group 4 FAX TIFF or monochrome PNG?

To render the XPS as a monochrome (1 bit per pixel) image compressed using G4 CCITT, simply add the option '--mono' within the command-line string. For example,

```
xpsconvert --mono -f TIFF --verb 10 --dpi 300 in.xps
```

To enable dithering (grayscale color simulation) add '--dither' option to the command line string.

3.3.16 Can I use XPSConvert for XAML to PDF conversion?

XPSConvert can be used to for XAML to PDF conversion. To convert XAML to PDF, first serialize flow document as XPS using specific styling information (e.g. page dimensions, header, footer, margins, etc), then pass the output to XPSConvert XPS to PDF conversion function.

In an application that uses Windows Presentation Foundation (WPD), writing the application's data as an XPS Document is straightforward. Windows Presentation Foundation applications describe their graphical content using Extensible Application Markup Language (XAML) code. XPS Documents use a subset of this code as their markup. A WPF application can create an XPS document by calling a method that flattens the application XAML elements of the page into to the XPS page markup.
A managed code application can also create the individual parts of an XPS Document by using the XPS Document API in the System.Windows.Xps.Serialization namespace. This may be a better

option for retaining precise control over how the XPS Document is constructed or for adding additional metadata to the document. An application can also create an XPS Document directly, even if the application was not built using Windows Presentation Foundation. This can be done by creating the markup for the individual XPS Document parts, but the classes and methods described here that actually create the XPS Document only run in a managed code environment.

The XamlStreamToXps method from the sample code shown below can enable your WPF application to create an XPS Document. XamlStreamToXps accepts as its arguments the file name of the XPS Document to be created and a Stream containing XAML, which describes the document. XamlStreamToXps will read the input stream and create an XPS Document.

```
void XamlStreamToXps(
    Stream srcXamlStream, ParserContext context, string destXpsFile)
{
    XpsDocument document =
        new XpsDocument(destXpsFile, FileAccess.ReadWrite);
    XpsPackagingPolicy packagePolicy = new XpsPackagingPolicy(document);
    XpsSerializationManager serializationMgr =
        new XpsSerializationManager(packagePolicy, false);

    object parsedDocObject = (null == context) ?
        XamlReader.Load (srcXamlStream) :
        XamlReader.Load (srcXamlStream, context);

    serializationMgr.SaveAsXaml(parsedDocObject);
    document.Close();
}
void XamlFileToXps(string srcXamlFile, string destXpsFile)
{
   using(Stream fileStream = File.OpenRead(srcXamlFile))
   {
      ParserContext context = new ParserContext();
      context.BaseUri = new Uri(Directory.GetCurrentDirectory() + "//");
      XamlStreamToXps(fileStream, context, destXpsFile);
   }
}
```

While the default conversion from XAML to XPS is very easy to use, if you want more features, you need to write some code. For full code samples and discussion on how to serialize XAML to XPS, you may want to refer to the following resources:

http://msdn.microsoft.com/en-us/magazine/cc163664.aspx
http://www.pdftron.com/xpsconvert/forum.html

3.3.17 Is XPSConvert available as an SDK for integration with third party applications?

For developers who are looking for a software development component to integrate into their application, PDFTron also offers XPSConvert SDK, an easy-to-use, yet powerful software component for embedding into client and server based applications. XPSConvert SDK is available as a shared library and can be easily accessed from any programming language (including C#, VB.NET, C/C++, Java, VB6, Perl, Python, Ruby, Delphi, etc). XPSConvert is also available as an add-on module to PDFNet SDK.

3.3.18 Does XPSConvert have any dependencies on third party components/software?

XPSConvert is a completely stand-alone application and does not include any dependencies on third-party components or software.

3.4 General Usage Examples

Example 1. The simplest command line: Convert XPS to PDF.

Notes:

- The default output format is PDF.
- The '-o' (or --output) parameter is used to specify the output folder. If this option was not specified, all images would be stored in the current working folder.

```
xpsconvert -o ex1 test/tiger.xps
```

Example 2. Convert XPS to PNG

- By default all images will be rendered at 92 Dots Per Inch (DPI).

```
xpsconvert –f png -o ex1 test/tiger.xps
```

Example 3. Convert XPS to JPEG at 300 DPI and higher compression.

Notes:

- The '-d' (or --dpi) parameter is used to specify the output image resolution.
- The '-f' (or --format) parameter specifies that the output format is JPEG.
- The '--verb' option instructs XPSConvert to output more feedback in the console window.

```
xpsconvert --output ex2 -d 300 -f jpg --verb 2 --quality 60 test/tiger.xps
```

Example 4. Convert XPS to a TIFF file of given pixel dimensions.

Notes:

- In this example, the '--hres' and '--vres' parameters are used instead of the '–dpi' parameter to specify absolute dimensions of the target image.

```
xpsconvert -o ex3 --hres 1000 --vres 1000 -f tif --verb 2
test/blue_secret.xps
```

Example 5. Convert XPS to multi-page TIFF.

To convert a XPS file to a multi-page TIFF, you can use the following lines:

```
xpsconvert -o OUT2 --multipage -f tif --gray --verb 2 D:\
xpsconvert --subfolders -o OUT2 --multipage -f tif --verb 3 "D:\My XPS"
```

3.5 Batch Processing and the Use of Wildcards

XPSConvert supports processing of multiple input documents in the same run. For example, it is possible to specify multiple XPS folders and XPSConvert will automatically process all XPS documents matching a given file extension. For example, the following command-line will process all XPS documents in folders ‘test1’ and ‘test2’

```
c:\> xpsconvert -o c:/output_folder c:/test1 c:/test2
```

Wildcard characters can also be used to process multiple input files.

For example, if a directory contains the following XPS documents:

```
C:\test1 >dir
 Directory of C:\test1
01/04/2007  03:35 PM    <DIR>          .
01/04/2007  03:35 PM    <DIR>          ..
05/21/2004  02:27 PM             A1.xps
05/03/2005  09:38 AM             A2.xps
05/20/2003  08:46 AM             B1.xps
05/15/2003  12:50 PM             B2.xps
```

To process all XPS documents in this folder, you could specify:

```
c:\>xpsconvert -o c:/output_folder c:/test1/*.xps
```

To process all XPS documents staring with ‘A’, you could specify:

```
xpsconvert -o c:/output_folder c:/test1/A*.xps
```

Or to process all XPS documents ending with ‘1’, you could specify:

```
xpsconvert -o c:/output_folder c:/test1/*1.xps
```

You can use either of the two standard wildcards — the question mark (?) and the asterisk (*) — to specify filename and path arguments on the command line.

The wildcards are expanded in the same manner as operating system commands. (Please refer to your operating system user's guide if you are unfamiliar with wildcards). Enclosing an argument in double quotation marks (" ") suppresses the wildcard expansion. Within quoted arguments, you can represent quotation marks literally by preceding the double-quotation-mark character with a backslash (****). If no matches are found for the wildcard argument, the argument is passed literally.

3.6 Exit Codes

To provide additional feedback, XPSConvert returns exit codes after completing processing. The exit codes can be used to provide user feedback, for logging etc. This is particularly important for applications running in an unattended environment.

The following table lists possible exit codes and their description:

Exit Code	Description
0	All files converted successfully.
1	Unspecified error.
2	Bad license key
3	Failed to create the output directory
4	Bad input filename or path

All codes other then ‘0’ indicate that there was an error during the conversion process.

The following illustrates a sample Windows batch script that processes exit codes:

```
@echo off rem convert all XPS files in ‘data’ folder
XPSConvert ./data
if errorlevel 1 goto othererror
if errorlevel 4 goto inputerr
if errorlevel 0 goto exit

:inputerr
echo No input files specified.
goto exit

:othererror
echo An error encountered during processing.
goto exit

:exit
```

4. Support

4.1 Reporting Problems

If you encounter a problem or question regarding PDFTron XPSConvert, which is not addressed on PDFTron's website, please submit a problem report to PDFTron's Support group at http://www.pdftron.com/reportproblem.html.

When submitting a problem you will be asked to provide the following information:

- Contact details
- Product and Version of the product
- Detailed description of problem
- Problem file(s)
- Whether you have an AMS (Annual Maintenance Subscription)
- Any other information that may be related

4.2 Contact Information

To contact PDFTron directly, please use the contact information below:

Tel: 1-604-730-8989
Fax: 1-604-676-2477

Web site: www.pdftron.com

Email Contacts:

General Business Inquiries: info@pdftron.com
Sales & Licensing: sales@pdftron.com
Product Support: support@pdftron.com
Professional Services: services@pdftron.com
Website related questions: webmaster@pdftron.com
Press & News: press@pdftron.com