

Release Notes on PDFsharp 1.2 and MigraDoc

PDFSHARP

PDFsharp was originally designed to work on top of GDI+ from namespace System.Drawing. The current beta version includes a build of PDFsharp that uses WPF instead of GDI+. We created this variation for a better integration in your WPF applications. The WPF based version of PDFsharp is still beta and should not be used in production code.

MigraDoc and most samples can now be compiled with PDFsharp based on GDI+ or based on WPF.

PDF AND XPS

This release contains a preview version of our forthcoming XPS to PDF converter. The converter transforms XPS files into equivalent PDF files. The preview version currently consists of a set of unit test functions that can be started with Visual Studio 2008 Professional or higher. The code is still in an early state.

MIGRADOCS

Aside from some minor bug fixes MigraDoc is unchanged in functionality. Because rendering (especially of tables) is rather slow we started to recode its internal data structures to increase the performance. This optimized version is not part of this release because it simply does not work until it is complete. We will release it when it is finished.

THE FOLDERS

The **bin** folder contains the compiled assemblies (Release Build) for PDFsharp and MigraDoc. We provide them as a service for people who do neither use Visual Studio 2005 nor Visual Studio 2008.

The **PDFsharp** folder contains PDFsharp (a detailed description of the sub-folders follows below).

The **MigraDoc** folder contains MigraDoc (a detailed description of the sub-folders follows below).

Folder list of PDFsharp

code

This folder contains the source code folders of PDFsharp and additional helper libraries. If you want to use PDFsharp in your projects, you only need one or more projects from this folder.

code/PdfSharp

The source code of PDFsharp. There are three projects:

- **PdfSharp.csproj**
PDFsharp for .NET 2.0 based on GDI+ (System.Drawing namespace). Previously this was the only project for PDFsharp.
- **PdfSharp-WPF.csproj**
PDFsharp for .NET 3.0 based on WPF (System.Windows namespace). This build of PDFsharp uses WPF instead of GDI+. This is new stuff and still in beta state.
- **PdfSharp-Hybrid.csproj**
Internal build used for testing only. This build is not intended to be used in projects.

code/PdfSharp.AcroPdfLib

A simple COM wrapper for Adobe Acrobat DLL. May be useful for viewing and printing PDF documents when Adobe Reader is installed. This library uses only functionality from Adobe, PDFsharp can neither render nor print PDF documents.

Please note: this library provides access to 3rd party libraries that are not included with PDFsharp.

code/PdfSharp.Charting

A library for drawing charts like line charts, bar charts, pie charts etc. This library is also used by MigraDoc.

code/PdfSharp.Ghostscript

A simple wrapper for Ghostscript. May be useful for creating bitmaps or thumbnails from a PDF document. This library uses only functionality from Ghostscript. PDFsharp cannot render PDF documents into bitmaps.

Please note: this library provides access to 3rd party libraries that are not included with PDFsharp.

code/PdfSharp.Xps

Beta version of the XPS to PDF conversion library. This library can read XPS files and convert them into PDF files.

dev

This folder contains the source code of applications used for developing and testing PDFsharp. There are the following solutions:

- **PDFsharp-VS2005.sln**
A solution for Visual Studio 2005 that builds PDFsharp for .NET 2.0 based on GDI+ and the corresponding projects.
- **PDFsharp-VS2008.sln**
A solution for Visual Studio 2008 that builds PDFsharp for .NET 2.0 based on GDI+ and the corresponding projects.
- **PdfSharp-WPF.sln**
A solution for Visual Studio 2008 that builds PDFsharp for .NET 3.0 based on WPF and the corresponding projects.

dev/PdfSharp.Charting.Demo

A demo program for the PDFsharp charting library.

dev/PdfSharp.Explorer

An application that shows the internal structures of a PDF file.

Tip: If you load a PDF file with the name **myfile.pdf** this program automatically creates a new file **myfile_.pdf**. This new file is a valid PDF file, but internally reformatted to be more “readable” for humans. All objects and dictionaries are indented and content streams are decompressed. You can load this file in a text editor and it is much more readable than the original file.

dev/PdfSharp.Ghostscript.Demo

A demo program that shows how to use the Ghostscript wrapper. You need Ghostscript to run this sample.

dev/PdfViewer.Demo

Displays a PDF file in a window by using Adobe Reader or Acrobat as an ActiveX control.

This program does not work under Vista. If anybody fixes this please let me know!

dev/XGraphicLab

A test program used for developing the GDI+ version of PDFsharp.

dev/XGraphicLab-WPF

A test program used for developing the WPF version of PDFsharp.

dev/XPStoPDF

Contains a solution that runs unit tests of the XPS to PDF converter. Requires Visual Studio 2008 Professional or higher for running the tests.

samples

This folder contains small sample programs that demonstrate the features of PDFsharp. This is the best resource for studying how PDFsharp works.

samples/PDFs

Some PDF files used in the samples.

samples/Samples C#

Samples written in C#.

samples/Samples C#/Based on GDI+

Samples that use PDFsharp for GDI+.

samples/Samples C#/Based on GDI+/Annotations

This sample shows how to create annotations.

samples/Samples C#/Based on GDI+/Booklet

This sample shows how to produce a booklet by placing two pages of an existing document on one landscape orientated page of a new document.

samples/Samples C#/Based on GDI+/Bookmarks

This sample shows how to create bookmarks. Bookmarks are called outlines in the PDF reference manual, that's why you deal with the class PdfOutline.

samples/Samples C#/Based on GDI+/ColorsCMYK

This sample shows how to use CMYK colors.

samples/Samples C#/Based on GDI+/CombineDocuments

This sample shows how to create a new document from two existing PDF files. The pages are inserted alternately from two external documents. This may be useful for visual comparison.

samples/Samples C#/Based on GDI+/ConcatenateDocuments

This sample shows how to concatenate the pages of several PDF documents to one single file.

samples/Samples C#/Based on GDI+/ExportImages

This sample shows how to export JPEG images from a PDF file.

samples/Samples C#/Based on GDI+/Graphics

This sample shows some of the capabilities of the XGraphics class.

samples/Samples C#/Based on GDI+/HelloWorld

This sample is the obligatory Hello World program.

samples/Samples C#/Based on GDI+/PageSizes

This sample shows a document with different page sizes. Note: You can set the size of a page to any size using the Width and Height properties. This sample just shows the predefined sizes.

samples/Samples C#/Based on GDI+/Preview

This sample shows how to render graphics in both a preview and a PDF document.

samples/Samples C#/Based on GDI+/PrintPdfFile

This sample shows how to print a PDF file automatically using Adobe Reader or Adobe Acrobat. PDFsharp cannot print PDF files, but you can use Adobe Reader with a command line switch to do the job.

samples/Samples C#/Based on GDI+/ProtectDocument

This sample shows how to protect a document with a password.

samples/Samples C#/Based on GDI+/SplitDocument

This sample shows how to convert a PDF document with n pages into n documents with one page each.

samples/Samples C#/Based on GDI+/TextLayout

This sample shows how to layout text with the TextFormatter class. TextFormatter is new since PDFsharp 0.9 and was provided because it was one of the "most wanted" features. But it is better and easier to use MigraDoc to format paragraphs. See also MigraDoc sample MixMigraDocAndPdfSharp.

samples/Samples C#/Based on GDI+/TwoPagesOnOne

This sample shows how to place two pages of an existing document on one landscape orientated page of a new document.

samples/Samples C#/Based on GDI+/Unicode

This sample shows how to use Unicode text in PDFsharp.

samples/Samples C#/Based on GDI+/UnprotectDocument

This sample shows how to unprotect a document (if you know the password). Note that we will not explain nor give any tips how to crack a protected document with PDFsharp.

samples/Samples C#/Based on GDI+/Watermark

This sample shows three variations how to add a watermark text to an existing PDF file.

samples/Samples C#/Based on GDI+/WorkOnPdfObjects

PDF documents are based internally on objects like dictionaries, arrays, streams etc. This sample shows how to work directly on these underlying PDF objects. Use this functionality to achieve PDF features that are not yet implemented in PDFsharp, e.g. adding an 'open action' to a document.

samples/Samples C#/Based on GDI+/XForms

This sample shows how to create an XForm object from scratch. You can think of such an object as a template, that, once created, can be drawn frequently anywhere in your PDF document.

samples/Samples C#/Based on WPF

Samples that use PDFsharp for WPF.

from samples/Samples C#/Based on WPF/Annotations
to samples/Samples C#/Based on WPF/XForms

Same samples as under **samples/Samples C#/Based on GDI+** but compile with PDFsharp based on WPF.

samples/Samples VB

Sample written in VB.net.

samples/Samples VB/HelloWorld

A “Hello, world” program written in VB.

samples/WebSamples

Sample written for ASP.NET.

samples/WebSamples/Clock

An ASP.NET application that creates a PDF with a clock that shows the current server time.

testing/

The only way I know to check whether a PDF file shows the correct content is to look at it. Most tests render their output to both a bitmap and to a PDF file. Then the result is combined in a PDF file called **~Result.pdf**. The left side of each result page shows the PDF output and the right side the bitmap.

You need Visual Studio 2008 Professional (or higher) to run the tests.

testing/PdfSharp.UnitTests

Tests for PDFsharp.

testing/PdfSharp.UnitTests/PDFsharpTests-GDI

A test solution that draws graphic primitives on both a bitmap (using GDI+) and a PDF page and then combines both drawings on a PDF page so that they can be compared visually.

testing/PdfSharp.UnitTests/PDFsharpTests-WPF

A test solution that draws graphic primitives on both a bitmap (using WPF) and a PDF page and then combines both drawings on a PDF page so that they can be compared visually.

If you run a test alone, none of the tests fail. But if you run all tests some of the tests fail because of a threading problem I do not understand yet.

testing/PdfSharp.Xps.UnitTests

Still in Beta state: Tests for converting XPS to PDF.

testing/SampleXpsDocuments_1_0

Empty folder for sample XPS documents. You must download the file **SampleXpsDocuments_1_0.exe** from <http://www.microsoft.com/whdc/XPS/XpsSamples.mspx>. The current beta version of PDFsharp cannot convert all XPS files without errors to PDF files.

Folder list of MigraDoc

code

This folder contains the source code folders of MigraDoc Lite. If you want to use MigraDoc in your projects, you only need the projects from this folder.

code/MigraDoc.DocumentObjectModel

The source code of MigraDoc's document object model.

code/MigraDoc.Rendering

The source code of MigraDoc's PDF document renderer.

code/MigraDoc.RtfRendering

The source code of MigraDoc's RTF document renderer.

dev

Solutions that build the MigraDoc projects.

samples

This folder contains small sample programs that demonstrate the features of MigraDoc. This is the best resource for studying how MigraDoc works.

samples/Samples C#

Samples written in C#.

samples/Samples C#/Based on GDI+

Samples that use MigraDoc which is based on PDFsharp for GDI+.

samples/Samples C#/Based on GDI+/DocumentViewer

This sample shows how to create a MigraDoc document and preview it in a window. It also demonstrates how to convert a MigraDoc document into a PDF file, RTF file, bitmap image, and Windows metafile.

samples/Samples C#/Based on GDI+/HelloMigraDoc

This sample shows how to create a MigraDoc document with paragraphs, images, tables, etc. and convert it into PDF.

samples/Samples C#/Based on GDI+/HelloWorld

This sample is the obligatory Hello World program for MigraDoc documents.

samples/Samples C#/Based on GDI+/Invoice

This sample shows how to create a simple invoice of a fictional book store. The invoice document is created with the MigraDoc document object model and then rendered to PDF.

samples/Samples C#/Based on GDI+/MixMigraDocAndPdfSharp

This sample demonstrates how to mix MigraDoc and Pdfsharp.

samples/Samples C#/Based on WPF

Samples that uses MigraDoc which is based on Pdfsharp for WPF.

from samples/Samples C#/Based on WPF/HelloMigraDoc
to samples/Samples C#/Based on WPF/Invoice

Same samples as under **samples/Samples C#/Based on GDI+** but compile with Pdfsharp based on WPF.

testing

Tests for MigraDoc.