

Super Pdf2Image Converter

.NET Edition

Version 1.5

Image and PDF viewer control for .NET applications

© 2009-2011 Software Siglo XXI
www.softwaresigloXXI.com

REFERENCE GUIDE

TABLE OF CONTENTS

Introduction.....	3
Version history.....	4
Installation and Deployment.....	5
Method Reference.....	6
Sample C# Code	7

Introduction

Super Pdf2Image Converter allows you to convert any pages from a PDF file to a rasterized image in just two lines of code, easy and quickly.

Super Pdf2Image Converter has been designed to be very easy to use. It's available as a very simple .NET library, written in C#, so you only need to call a function to get the rasterized image of any page of a PDF file.

Typical usage includes:

- Create thumbnails of any page of a PDF file.
- Display PDF files from your WinForm or ASP.NET application.
- Conversion of PDF files to raster/bitmap images.
- Batch conversion.

Feature List:

- ❖ Designed to minimize complexity - only 2 lines of code needed.
- ❖ Fast page count and size detect.
- ❖ Can specify page number to rasterize, resolution, image format, width and height.
- ❖ Native 100% C# component (instant use in C# or VB.NET projects).
- ❖ Requires GhostScript library.
- ❖ Easy to distribute: only 1 DLL (component + compressed dictionaries)

Output file formats supported:

- ✓ PNG (24-bit color/truecolor or grayscale)
- ✓ JPEG
- ✓ BMP (8-bit color and truecolor)
- ✓ TIFF (24-bit color, 12-bit color, grayscale and bitonal: CCITT G3/G4, 2D G3, LZW, Packbits)

If you need further help about anything, don't hesitate to contact us at:
contact@softwaresigloxxi.com or visit our website at www.softwareSigloXXI.com.

Version history

Version 1.5

- Added support 64 bit CPU

Version 1.4

- Added support for CMYK images
- Improved page size detection
- Many minor bugs fixed

Version 1.0

- First release

Installation and Deployment

The Pdf2Image library is provided as a single DLL (Pdf2Image.DLL).

You need also GhostScript library installed on the computer. You may obtain a copy of this library for free here:

<http://sourceforge.net/projects/ghostscript/>

Download latest release and copy the file `gsdll32.dll` or `gsdll64.dll` to the Windows/System32 folder.

Then, to use the library in your application simply add a reference to it from your VB.NET or C# project.

To deploy the control, include Pdf2Image.dll along your application main or /bin directory. You must also install SourceForge in destination computer (or make sure gsdll32.dll or gsdll64.dll is copied to the Windows/System32 folder).

Method Reference

Super Pdf2Image Converter includes the following public methods:

- **GetPageCount()**
Returns the page count of any PDF file.
- **GetPageSize(pageNumber, out pageWidth, out pageHeight)**
Returns the page size in original PDF coordinates.
- **GetImage(pageNumber, resolution, imageFormat)**
Returns specified page of PDF file as an Image object. Original size is maintained.
- **GetImage(pageNumber, resolution, width, height, imageFormat)**
Returns specified page of PDF file as an Image object. Original image is resized and resampled to fit width/height
- **GetImage(outputFileName, firstPage, lastPage, resolution, imageFormat)**
Converts specifies pages to image and save them to outputFileName (tiff allows multi-page or creates several files)

Sample C# Code

```
// Instantiate the component
```

```
Pdf2ImageConverter p2i = new Pdf2ImageConverter(pdfPath);
```

```
// Get page count of a PDF file
```

```
int pages = p2i.GetPageCount();
```

```
// Get size of any page
```

```
int width, height;
```

```
p2i.GetPageSize(1, out width, out height);
```

```
// Convert any page of PDF to image (returns bitmap object with original size)
```

```
Bitmap bm = p2i.GetImage(pageNumber, resolution, imageFormat);
```

```
// Convert any page of PDF to image (returns bitmap object with custom size)
```

```
Bitmap bm = p2i.GetImage(pageNumber, resolution, width, height, imageFormat);
```

```
// Convert any page of PDF to image file (preserving aspect ratio)
```

```
p2i.GetImage(outputImagePath, pageNumber, resolution, imageFormat);
```