

Siberix PDF Sparkle

User's Guide

User's Guide for Siberix PDF Sparkle 2.0

© 2003-2012 Siberix Technologies, Canada. All rights reserved.

The information in this document is provided "AS-IS", with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. Furthermore, information provided in this document is preliminary, and may be changed. This document is provided for informational purposes only. Siberix Technologies disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information presented in this document. Siberix Technologies does not warrant or represent that such use will not infringe such rights. Siberix Technologies retains the right to make changes to this specification at any time, without notice.

Contents

Overview.....	4
Top Features.....	4
System Requirements.....	4
Installation.....	5
Graphics.....	6
Measuring.....	6
Colors.....	6
Pens.....	7
Brushes.....	8
SolidColorBrush.....	8
LinearGradientBrush.....	8
RadialGradientBrush.....	9
TextureBrush.....	9
Fonts.....	10
Images.....	10
Document.....	11
ASPX Integration.....	12

Overview

Siberix PDF Sparkle is a top-notch C# library for .NET 2.0 - 4.0 designed to create industry standard PDF documents dynamically with API calls. It provides an interface similar to the System.Drawing.Graphics class for every page of the PDF document being created.

Top Features

- PDF publishing
- Easy to use API (similar to GDI+)
- Hyperlinks and Bookmarks
- PDF viewer preferences
- Custom PDF properties
- 128-bit Encryption
- Image alpha channels
- Clippings and transformations
- RGB, CMYK and Grey PDF color schemes
- Pen styles, widths, colors, caps and joins
- Path composition, Bezier curves and Splines
- Lines, Rectangles, Ellipses, Polylines, Polygons, Arcs and Pies
- SolidColor, LinearGradient, RadialGradient and Texture brushes

System Requirements

- PC with an Intel or compatible Pentium 600 MHz or higher processor
- Windows XP, Windows Server 2008, Windows Vista, Windows 7
- .NET Framework 2.0 - 4.0
- Minimum of 256 Mb of RAM (512 and more recommended)
- Memory requirements vary according to the nature of the project

Installation

Siberix PDF Sparkle doesn't require any installation. All you need to do is to copy the [Siberix.PDFSparkle.dll](#) into your project's folder (desktop applications) or to the "bin" folder (ASPX applications) and make a reference to it.

Graphics

This namespace provides core drawing functionality.

Measuring

The default unit of measure for most of the elements described in this User's Guide is point (pt). You may find some useful expressions below.

- 1 pt = 1 / 72 inch
- 1 px = 1 / 96 inch - May vary for particular screen devices and modes
- 1 mm = 1 / 25.4 inch

Colors

Colors are widely used across the document. Every time you create a pen or a brush you need to provide a reference to a color object. Siberix PDF Sparkle allows to specify colors in two different ways: 1. You can create a new color providing Red, Green, Blue and Alpha byte values; 2. You can select a color by its name from a collection of predefined colors. To access the collection of predefined colors use properties of the [Siberix.Sparkle.Graphics.Colors](#) class.

	AliceBlue		DarkGreen		Gray		LightYellow
	AntiqueWhite		DarkKhaki		Green		Lime
	Aqua		DarkMagenta		GreenYellow		LimeGreen
	Aquamarine		DarkOliveGreen		Honeydew		Linen
	Azure		DarkOrange		HotPink		Magenta
	Beige		DarkOrchid		IndianRed		Maroon
	Bisque		DarkRed		Indigo		MediumAquamarine
	Black		DarkSalmon		Ivory		MediumBlue
	BlanchedAlmond		DarkSeaGreen		Khaki		MediumOrchid
	Blue		DarkSlateBlue		Lavender		MediumPurple
	BlueViolet		DarkSlateGray		LavenderBlush		MediumSeaGreen
	Brown		DarkTurquoise		LawnGreen		MediumSlateBlue
	BurlyWood		DarkViolet		LemonChiffon		MediumSpringGreen
	CadetBlue		DeepPink		LightBlue		MediumTurquoise
	Chartreuse		DeepSkyBlue		LightCoral		MediumVioletRed
	Chocolate		DimGray		LightCyan		MidnightBlue
	Coral		DodgerBlue		LightGoldenrodYellow		MintCream
	CornflowerBlue		Firebrick		LightGray		MistyRose
	Cornsilk		FloralWhite		LightGreen		Moccasin
	Crimson		ForestGreen		LightPink		NavajoWhite
	Cyan		Fuchsia		LightSalmon		Navy
	DarkBlue		Gainsboro		LightSeaGreen		OldLace
	DarkCyan		GhostWhite		LightSkyBlue		Olive
	DarkGoldenrod		Gold		LightSlateGray		OliveDrab
	DarkGray		Goldenrod		LightSteelBlue		Orange

 OrangeRed	 Plum	 SeaShell	 Teal
 Orchid	 PowderBlue	 Sienna	 Thistle
 PaleGoldenrod	 Purple	 Silver	 Tomato
 PaleGreen	 Red	 SkyBlue	 Turquoise
 PaleTurquoise	 RosyBrown	 SlateBlue	 Violet
 PaleVioletRed	 RoyalBlue	 SlateGray	 Wheat
 PapayaWhip	 SaddleBrown	 Snow	 White
 PeachPuff	 Salmon	 SpringGreen	 WhiteSmoke
 Peru	 SandyBrown	 SteelBlue	 Yellow
 Pink	 SeaGreen	 Tan	 YellowGreen

Examples

[CS]

```
Siberix.Sparkle.Graphics.Color color1 = new Siberix.Sparkle.Graphics.Color(255, 0, 0);
Siberix.Sparkle.Graphics.Color color2 = new Siberix.Sparkle.Graphics.Color(50, 255, 0, 0);
Siberix.Sparkle.Graphics.Color color3 = Siberix.Sparkle.Graphics.Colors.Red;
```

[VB]

```
Dim color1 As Siberix.Sparkle.Graphics.Color = New Siberix.Sparkle.Graphics.Color(255, 0, 0)
Dim color2 As Siberix.Sparkle.Graphics.Color = New Siberix.Sparkle.Graphics.Color(50, 255, 0, 0)
Dim color3 As Siberix.Sparkle.Graphics.Color = Siberix.Sparkle.Graphics.Colors.Red
```

Pens

Every time you draw a line or a curve you have to provide a reference to a pen object. Siberix PDF Sparkle allows to set individual pen attributes including Color, Width, Style, Cap, Join and MiterLimit.

Predefined collection

You may use a collection of predefined pens to ease the development within your code. To access the collection of predefined pens use properties of the [Siberix.Sparkle.Graphics.Pens](#) class.

Examples

[CS]

```
Siberix.Sparkle.Graphics.Pen pen1 = new Siberix.Sparkle.Graphics.Pen(Siberix.Sparkle.Graphics.Colors.Red);
Siberix.Sparkle.Graphics.Pen pen2 = new Siberix.Sparkle.Graphics.Pen(new Siberix.Sparkle.Graphics.Color(255, 0, 0), 2,
Siberix.Sparkle.Graphics.DashStyle.DashDot);
Siberix.Sparkle.Graphics.Pen pen3 = Siberix.Sparkle.Graphics.Pens.Red;
```

[VB]

```
Dim pen1 As Siberix.Sparkle.Graphics.Pen = New Siberix.Sparkle.Graphics.Pen(Siberix.Sparkle.Graphics.Colors.Red)
Dim pen2 As Siberix.Sparkle.Graphics.Pen = New Siberix.Sparkle.Graphics.Pen(New Siberix.Sparkle.Graphics.Color(255, 0, 0), 2,
Siberix.Sparkle.Graphics.DashStyle.DashDot)
Dim pen3 As Siberix.Sparkle.Graphics.Pen = Siberix.Sparkle.Graphics.Pens.Red
```

Brushes

Siberix PDF Sparkle provides several types of brushes including SolidColorBrush, LinearGradientBrush, RadialGradientBrush and TextureBrush to output text or to fill the interiors of graphical shapes.

SolidColorBrush

Predefined collection

You may use a collection of predefined brushes to ease the development within your code. To access the collection of predefined brushes use properties of the [Siberix.Sparkle.Graphics.Brushes](#) class.

Examples

[CS]

```
Siberix.Sparkle.Graphics.SolidColorBrush brush1 = new
Siberix.Sparkle.Graphics.SolidColorBrush(Siberix.Sparkle.Graphics.Colors.Red);

Siberix.Sparkle.Graphics.SolidColorBrush brush2 = new Siberix.Sparkle.Graphics.SolidColorBrush(new
Siberix.Sparkle.Graphics.Color(255, 0, 0));

Siberix.Sparkle.Graphics.Brush brush3 = Siberix.Sparkle.Graphics.Brushes.Red;
```

[VB]

```
Dim brush1 As Siberix.Sparkle.Graphics.SolidColorBrush = New
Siberix.Sparkle.Graphics.SolidColorBrush(Siberix.Sparkle.Graphics.Colors.Red)

Dim brush2 As Siberix.Sparkle.Graphics.SolidColorBrush = New Siberix.Sparkle.Graphics.SolidColorBrush(New
Siberix.Sparkle.Graphics.Color(255, 0, 0))

Dim brush3 As Siberix.Sparkle.Graphics.Brush = Siberix.Sparkle.Graphics.Brushes.Red
```

LinearGradientBrush

Examples

[CS]

```
Siberix.Sparkle.Graphics.LinearGradientBrush brush1 = new
Siberix.Sparkle.Graphics.LinearGradientBrush(Siberix.Sparkle.Graphics.Colors.Red, Siberix.Sparkle.Graphics.Colors.Yellow);

Siberix.Sparkle.Graphics.ColorBlend blend = new Siberix.Sparkle.Graphics.ColorBlend();
blend.Add(Siberix.Sparkle.Graphics.Colors.Red, 0);
blend.Add(Siberix.Sparkle.Graphics.Colors.Yellow, 0.5F);
blend.Add(Siberix.Sparkle.Graphics.Colors.Blue, 1);

Siberix.Sparkle.Graphics.LinearGradientBrush brush2 = new Siberix.Sparkle.Graphics.LinearGradientBrush(blend, 35);
```

[VB]

```
Dim brush1 As Siberix.Sparkle.Graphics.LinearGradientBrush = New
Siberix.Sparkle.Graphics.LinearGradientBrush(Siberix.Sparkle.Graphics.Colors.Red, Siberix.Sparkle.Graphics.Colors.Yellow)

Dim blend As Siberix.Sparkle.Graphics.ColorBlend = New Siberix.Sparkle.Graphics.ColorBlend()
blend.Add(Siberix.Sparkle.Graphics.Colors.Red, 0)
blend.Add(Siberix.Sparkle.Graphics.Colors.Yellow, 0.5F)
blend.Add(Siberix.Sparkle.Graphics.Colors.Blue, 1)

Dim brush2 As Siberix.Sparkle.Graphics.LinearGradientBrush = New Siberix.Sparkle.Graphics.LinearGradientBrush(blend, 35)
```


RadialGradientBrush

Examples

[CS]

```
Siberix.Sparkle.Graphics.RadialGradientBrush brush1 = new
Siberix.Sparkle.Graphics.RadialGradientBrush(Siberix.Sparkle.Graphics.Colors.Red, Siberix.Sparkle.Graphics.Colors.Yellow);

Siberix.Sparkle.Graphics.ColorBlend blend = new Siberix.Sparkle.Graphics.ColorBlend();
blend.Add(Siberix.Sparkle.Graphics.Colors.Red, 0);
blend.Add(Siberix.Sparkle.Graphics.Colors.Yellow, 0.5F);
blend.Add(Siberix.Sparkle.Graphics.Colors.Blue, 1);

Siberix.Sparkle.Graphics.Point center = new Siberix.Sparkle.Graphics.Point(0.4F, 0.6F);
Siberix.Sparkle.Graphics.Point radius = new Siberix.Sparkle.Graphics.Point(0.8F, 0.4F);
Siberix.Sparkle.Graphics.Point focus = new Siberix.Sparkle.Graphics.Point(0.7F, 0.2F);

Siberix.Sparkle.Graphics.RadialGradientBrush brush2 = new Siberix.Sparkle.Graphics.RadialGradientBrush(blend, 35, center,
radius, focus);
```

[VB]

```
Dim brush1 As Siberix.Sparkle.Graphics.RadialGradientBrush = New
Siberix.Sparkle.Graphics.RadialGradientBrush(Siberix.Sparkle.Graphics.Colors.Red, Siberix.Sparkle.Graphics.Colors.Yellow)

Dim blend As Siberix.Sparkle.Graphics.ColorBlend = New Siberix.Sparkle.Graphics.ColorBlend()
blend.Add(Siberix.Sparkle.Graphics.Colors.Red, 0)
blend.Add(Siberix.Sparkle.Graphics.Colors.Yellow, 0.5F)
blend.Add(Siberix.Sparkle.Graphics.Colors.Blue, 1)

Dim center As Siberix.Sparkle.Graphics.Point = New Siberix.Sparkle.Graphics.Point(0.4F, 0.6F)
Dim radius As Siberix.Sparkle.Graphics.Point = New Siberix.Sparkle.Graphics.Point(0.8F, 0.4F)
Dim focus As Siberix.Sparkle.Graphics.Point = New Siberix.Sparkle.Graphics.Point(0.7F, 0.2F)

Dim brush2 As Siberix.Sparkle.Graphics.RadialGradientBrush = New Siberix.Sparkle.Graphics.RadialGradientBrush(blend, 35,
center, radius, focus)
```

TextureBrush

Examples

[CS]

```
Siberix.Sparkle.Graphics.Image image = new Siberix.Sparkle.Graphics.Image("image.jpg");

Siberix.Sparkle.Graphics.TextureBrush brush = new Siberix.Sparkle.Graphics.TextureBrush(image);
```

[VB]

```
Dim image As Siberix.Sparkle.Graphics.Image = New Siberix.Sparkle.Graphics.Image("image.jpg")

Dim brush As Siberix.Sparkle.Graphics.TextureBrush = New Siberix.Sparkle.Graphics.TextureBrush(image)
```

Fonts

Siberix PDF Sparkle allows to output text using any valid True Type font. You may use a collection of internal font wrappers or provide a reference to a local TTF font file. Please note that you are only allowed to use fonts that permit document embedding or you have a correspondent license from the creator of the font.

Internal font wrappers

You may use a collection of internal font wrappers from the [Siberix.Sparkle.Graphics.Fonts](#) namespace. Please note that wrappers only provide a simplified way of referencing [System](#) available fonts.

- Arial
- ArialBold
- ArialBoldItalic
- ArialItalic
- CourierNew
- CourierNewBold
- CourierNewBoldItalic
- CourierNewItalic
- Garamond
- GaramondBold
- GaramondItalic
- Impact
- MonotypeCorsiva
- SansSerif
- Symbol
- Tahoma
- TahomaBold
- TimesNewRoman
- TimesNewRomanBold
- TimesNewRomanBoldItalic
- TimesNewRomanItalic
- Verdana
- VerdanaBold
- VerdanaBoldItalic
- VerdanaItalic
- Webdings
- Wingding

Examples

[CS]

```
Siberix.Sparkle.Graphics.Font font = new Siberix.Sparkle.Graphics.Fonts.ArialBold(12);  
Siberix.Sparkle.Graphics.Font font = new Siberix.Sparkle.Graphics.Font("font.ttf", 12);
```

[VB]

```
Dim font As Siberix.Sparkle.Graphics.Font = New Siberix.Sparkle.Graphics.Fonts.ArialBold(12)  
Dim font As Siberix.Sparkle.Graphics.Font = New Siberix.Sparkle.Graphics.Font("font.ttf", 12)
```

Images

You can use any type of raster images (BMP, JPG, GIF, TIFF, PNG, etc.) supported by .NET Framework 2.0 with the Siberix PDF Sparkle. However, we would recommend to use JPG images for photographic pictures and GIF or PNG images to preserve color and alpha transparency.

Examples

[CS]

```
Siberix.Sparkle.Graphics.Image image = new Siberix.Sparkle.Graphics.Image("image.jpg");
```

[VB]

```
Dim image As Siberix.Sparkle.Graphics.Image = New Siberix.Sparkle.Graphics.Image("image.jpg")
```

Document

Siberix PDF Sparkle is a very simple, yet powerful PDF generation tool. PDF document creation has never been that easy. You start with the root object for the PDF document, set info properties and security preferences, then you add pages to the root object and for every page you get an access to the IGraphics interface that allows to output text, images and graphical shapes. Along the way, you may build a bookmark tree and enrich the final PDF document with a set of hyperlinks.

Examples

[CS]

```
Siberix.Sparkle.PDF.Document document = new Siberix.Sparkle.PDF.Document();

document.Info.Title = "Example";
document.Info.Author = "Siberix Technologies, Canada";
document.Info.Creator = "Siberix PDF Sparkle for .NET";

document.User = "test";
document.Owner = "siberix";

document.Permissions.Add = false;
document.Permissions.Copy = false;
document.Permissions.Modify = false;
document.Permissions.Print = true;

Siberix.Sparkle.PDF.IPage page = document.AddPage(612, 792);

document.AddBookmark(0, 0, 150, "Text", true);

page.Graphics.Font = new Siberix.Sparkle.Graphics.Fonts.Arial(12);
page.Graphics.Brush = Siberix.Sparkle.Graphics.Brushes.Red;
page.Graphics.DrawString(10, 150, "Hello PDF");

document.Generate("Document.pdf");
```

[VB]

```
Dim document As Siberix.Sparkle.PDF.Document = New Siberix.Sparkle.PDF.Document()

document.Info.Title = "Example"
document.Info.Author = "Siberix Technologies, Canada"
document.Info.Creator = "Siberix PDF Sparkle for .NET"

document.User = "test"
document.Owner = "siberix"

document.Permissions.Add = False
document.Permissions.Copy = False
document.Permissions.Modify = False
document.Permissions.Print = True

Dim page As Siberix.Sparkle.PDF.IPage = document.AddPage(612, 792)

document.AddBookmark(0, 0, 150, "Text", True)

page.Graphics.Font = New Siberix.Sparkle.Graphics.Fonts.Arial(12)
page.Graphics.Brush = Siberix.Sparkle.Graphics.Brushes.Red
page.Graphics.DrawString(10, 150, "Hello PDF")

document.Generate("Document.pdf")
```

ASPX Integration

There are several ways to stream out PDF or XPS documents from ASPX pages.

Content type:

PDF - **application/pdf**

1. Streaming into the context's response.

[CS]

```
System.Web.HttpContext context = System.Web.HttpContext.Current;
System.Web.HttpResponse response = context.Response;
response.Clear();
response.ContentType = "application/pdf";
document.Generate(response.OutputStream);
response.End();
```

[VB]

```
Dim context As System.Web.HttpContext = System.Web.HttpContext.Current
Dim response As System.Web.HttpResponse = context.Response
response.Clear()
response.ContentType = "application/pdf"
document.Generate(response.OutputStream)
response.End()
```

Note: To stream out a PDF document over the **https** response please use the following lines:

[CS]

```
System.IO.MemoryStream ms = new System.IO.MemoryStream();

document.Generate(ms);

byte[] buffer = ms.ToArray();
int length = buffer.Length;
```

```
System.Web.HttpContext context = System.Web.HttpContext.Current;
System.Web.HttpResponse response = context.Response;
response.Clear();
response.ContentType = "application/pdf";
response.AddHeader("Content-Length", length.ToString());
response.AddHeader("Accept-Ranges", "bytes");
response.AddHeader("Accept-Header", length.ToString());
response.OutputStream.Write(buffer, 0, length);
response.End();
```

[VB]

```
Dim ms As System.IO.MemoryStream = New System.IO.MemoryStream()

document.Generate(ms)

Dim buffer As Byte() = ms.ToArray()
Dim length As Integer = buffer.Length

Dim context As System.Web.HttpContext = System.Web.HttpContext.Current
Dim response As System.Web.HttpResponse = context.Response
response.Clear()
response.ContentType = "application/pdf"
response.AddHeader("Content-Length", length.ToString())
response.AddHeader("Accept-Ranges", "bytes")
response.AddHeader("Accept-Header", length.ToString())
response.OutputStream.Write(buffer, 0, length)
response.End()
```

2. Streaming into the "Response" with the "inline" attribute.

[CS]

```
Response.Clear();
Response.ContentType = "application/pdf";
Response.AddHeader("content-disposition", "inline; filename=Document.pdf");
document.Generate(Response.OutputStream);
Response.End();
```

[VB]

```
Response.Clear()
Response.ContentType = "application/pdf"
Response.AddHeader("content-disposition", "inline; filename=Document.pdf")
document.Generate(Response.OutputStream)
Response.End()
```

3. Streaming into the "Response" with the "attachment" attribute.

[CS]

```
Response.Clear();
Response.ContentType = "application/pdf";
Response.AddHeader("content-disposition", "attachment; filename=Document.pdf");
document.Generate(Response.OutputStream);
Response.End();
```

[VB]

```
Response.Clear()
Response.ContentType = "application/pdf"
Response.AddHeader("content-disposition", "attachment; filename=Document.pdf")
document.Generate(Response.OutputStream)
Response.End()
```

4. Saving to the server and redirecting to some HTML page with the link to the generated PDF document.

[CS]

```
System.IO.FileStream stream = new System.IO.FileStream(Server.MapPath("Document.pdf"), System.IO.FileMode.Create,
System.IO.FileAccess.Write);
document.Generate(stream);
stream.Close();
```

```
Response.Redirect("link.html");
or
Response.Clear();
Response.Output.Write("<a href=\"Document.pdf\">PDF</a>");
Response.End();
```

[VB]

```
Dim stream As System.IO.FileStream = New System.IO.FileStream(Server.MapPath("Document.pdf"), System.IO.FileMode.Create,
System.IO.FileAccess.Write)
document.Generate(stream)
stream.Close()
```

```
Response.Redirect("link.html")
or
Response.Clear()
Response.Output.Write("<a href=\"\"Document.pdf\"\">PDF</a>");
Response.End()
```