Windows Phone中画笔（Brushes）

在这篇文章中，我们将要学习Windows Phone 7中所有有关Silverlight 画笔的（Brush）API，在windows phone 应用程序中画笔API用来填充颜色以及使用颜色或者图片设置背景。在这个例子中，我使用XAML编程实现API的功能。

中文

介绍

在这篇文章中我们将要学习在Windows Phone 7 的XAML 编程中如何使用Silverlight Brush API 。基本上Brush类允许你绘制屏幕的一部分区域、使用颜色或者图片刷设置屏幕背景颜色或者图片，除此之外我们还可以使用视频刷访问设备的摄像头。Brush 类有很多适用于Windows Phone 7的画笔。

画笔分类

在Windows Phone 7 SDK中，Silverlight支持5种不同类型的画笔和不同类型的输出。

1. SolidColorBrush
2. LinearGradientBrush
3. RadialGradientBrush
4. ImageBrush
5. VideoBrush

在这个例子中我使用panoramic页面实现绘制不同对象的功能（如矩形、圆、多边形、文本和页面背景）。

SolidColorBrush

纯色画笔（Solid Color Brush）API,通过设置一个0和1之间的值到opacity属性来绘制/填充一个具有透明色的单一颜色。其中0是完全透明，1是完全不透明。 在XAML 中创建一个SolidColorBrush，请看以下代码片断。

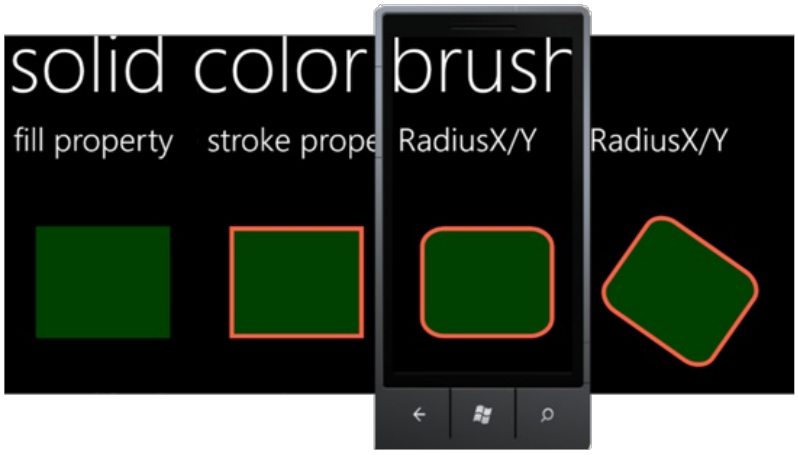
```
<SolidColorBrush Color="Green" Opacity="0.5" />
```

-PPSolidColorBrush.xaml

```
<Grid x:Name="LayoutRoot">
        <controls:Panorama Title="solid color brush">
            <!--Panorama item one-->
            <controls:PanoramaItem Header="fill property">
                <Grid>
                    <Rectangle Width="300" Height="250">
                        <Rectangle.Fill>
                            <SolidColorBrush Color="Green" Opacity="0.5" />
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item two-->
            <controls:PanoramaItem Header="stroke property">
                <Grid>
                    <Rectangle Width="300" Height="250" Stroke="Tomato" StrokeThickness="10">
                        <Rectangle.Fill>
                            <SolidColorBrush Color="Green" Opacity="0.5" />
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>

            </controls:PanoramaItem>

            <!--Panorama item three-->
            <controls:PanoramaItem Header="RadiusX/Y">
                <Grid>
                    <Rectangle Width="300" Height="250" Stroke="Tomato" StrokeThickness="10" RadiusX="50" RadiusY="40">
                        <Rectangle.Fill>
                            <SolidColorBrush Color="Green" Opacity="0.5" />
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>

            </controls:PanoramaItem>

            <!--Panorama item four-->
            <controls:PanoramaItem Header="RadiusX/Y">
                <Grid>
                    <Rectangle Width="300" Height="250" Stroke="Tomato" StrokeThickness="10" RadiusX="50" RadiusY="40">
                        <Rectangle.Fill>
                            <SolidColorBrush Color="Green" Opacity="0.5" />
                        </Rectangle.Fill>
                        <Rectangle.RenderTransform>
                            <RotateTransform Angle="35" CenterX="120" CenterY="140" />
                        </Rectangle.RenderTransform>
                    </Rectangle>
                </Grid>

            </controls:PanoramaItem>
        </controls:Panorama>
    </Grid>
```


(c) Copyright Nokia 2013. All rights reserved.

- XAML页面设计视图：



LinearGradientBrush

线性渐变画笔（Linear Gradient Brush）使用两个主要属性StartPoint（默认为"0,0"）和EndPoint （默认为"1,0"）来绘制两个以上的线性渐变颜色。 在XAML中创建一个LinearGradientBrush，请看以下代码片断。

```
<LinearGradientBrush StartPoint="0,0" EndPoint="1,0">
                                <GradientStop Color="Yellow" Offset="0.0" />
                                <GradientStop Color="Red" Offset="0.25" />
                                <GradientStop Color="Blue" Offset="0.75" />
                                <GradientStop Color="LimeGreen" Offset="1.0" />
                            </LinearGradientBrush>
```

- LinearGradientBrush.xaml

```
<Grid x:Name="LayoutRoot">
        <controls:Panorama Title="linear gradient brush">

            <!--Panorama item one-->
            <controls:PanoramaItem Header="horizontal">
                <Grid>
                    <Rectangle Width="300" Height="250">
                        <Rectangle.Fill>
                            <LinearGradientBrush StartPoint="0,0" EndPoint="1,0">
                                <GradientStop Color="Yellow" Offset="0.0" />
                                <GradientStop Color="Red" Offset="0.25" />
                                <GradientStop Color="Blue" Offset="0.75" />
                                <GradientStop Color="LimeGreen" Offset="1.0" />
                            </LinearGradientBrush>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item two-->
            <controls:PanoramaItem Header="vertical">
                <Grid>
                    <Rectangle Width="300" Height="250">
                        <Rectangle.Fill>
                            <LinearGradientBrush StartPoint="0,0" EndPoint="0,1">
                                <GradientStop Color="Yellow" Offset="0.0" />
                                <GradientStop Color="Red" Offset="0.25" />
                                <GradientStop Color="Blue" Offset="0.75" />
                                <GradientStop Color="LimeGreen" Offset="1.0" />
                            </LinearGradientBrush>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item three-->
            <controls:PanoramaItem Header="diagonal">
                <Grid>
                    <Rectangle Width="300" Height="250">
                        <Rectangle.Fill>
                            <LinearGradientBrush StartPoint="0,0" EndPoint="1,1">
                                <GradientStop Color="Green" Offset="0.0" />
                                <GradientStop Color="Yellow" Offset="0.25" />
                                <GradientStop Color="Blue" Offset="0.75" />
                                <GradientStop Color="Red" Offset="1.0" />
                            </LinearGradientBrush>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>
        </controls:Panorama>
    </Grid>
```

- XAML页面设计视图：

(C) Copyright Nokia 2013. All rights reserved.

RadialGradientBrush

径向渐变画笔（Radial Gradient Brush）用来绘制或者填充两个以上的圆形渐变 或者循环方式的颜色。 在XAML 中创建一个RadialGradientBrush，请看以下代码片断。

```
<RadialGradientBrush RadiusX=".5" RadiusY=".5" GradientOrigin=".0,.5" SpreadMethod="Reflect">
                                <GradientStop Color="Red" Offset="0.5" />
                                <GradientStop Color="BlueViolet" Offset="1" />
                            </RadialGradientBrush>
```

在这里: Radius x, Radius y: 这个属性用来设置渐变x轴半径和y轴半径。 Gradient Origin: 这个属性用来设置渐变的起始点。 Spread Method: 这个属性用来设置如何在要绘制对象的边界内绘制开始或结束的渐变。 **- RadialGradientBrush.xaml**

```
<Grid x:Name="LayoutRoot">
        <controls:Panorama Title="radial gradient brush">
            <!--Panorama item one-->
            <controls:PanoramaItem Header="rectangle">
                <Grid>
                    <Rectangle Width="300" Height="250" RadiusX="100" RadiusY="100">
                        <Rectangle.Fill>
                            <RadialGradientBrush RadiusX=".5" RadiusY=".5" GradientOrigin=".0,.5" SpreadMethod="Reflect">
                                <GradientStop Color="Red" Offset="0.5" />
                                <GradientStop Color="BlueViolet" Offset="1" />
                            </RadialGradientBrush>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item two-->
            <controls:PanoramaItem Header="rectangle">
                <Grid>
                    <Rectangle Width="300" Height="250">
                        <Rectangle.Fill>
                            <RadialGradientBrush>
                                <GradientStop Color="Red" Offset=".1"/>
                                <GradientStop Color="Blue" Offset=".2"/>
                                <GradientStop Color="Wheat" Offset=".3"/>
                                <GradientStop Color="Green" Offset=".4"/>
                                <GradientStop Color="Yellow" Offset=".6"/>
                                <GradientStop Color="BlueViolet" Offset="1"/>
                            </RadialGradientBrush>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item three-->
            <controls:PanoramaItem Header="ellipse">
                <Grid>
                    <Ellipse Height="300" Width="300">
                        <Ellipse.Fill>
                            <RadialGradientBrush>
                                <GradientStop Color="Red" Offset="0.0"/>
                                <GradientStop Color="LightPink" Offset="0.5"/>
                                <GradientStop Color="DarkGray" Offset="1.0"/>
                            </RadialGradientBrush>
                        </Ellipse.Fill>
                    </Ellipse>
                </Grid>
            </controls:PanoramaItem>

            <controls:PanoramaItem Header="ellipse">
                <Grid>
                    <Ellipse Height="300" Width="300">
                        <Ellipse.Fill>
                            <RadialGradientBrush>
                                <GradientStop Color="Red" Offset="0.0"/>
                                <GradientStop Color="Yellow" Offset="0.5"/>
                                <GradientStop Offset="1.0"/>
                            </RadialGradientBrush>
                        </Ellipse.Fill>
                    </Ellipse>
                </Grid>
            </controls:PanoramaItem>
```


(c) Copyright Nokia 2013. All rights reserved.

```
        </controls:Panorama>
    </Grid>
```

'**最重要**:' 默认的windows phone 7应用程序运行在16位。然而我们也可以使用32位颜色获取更好的色彩效果，在WMAppManifest.xml 中绑定渐变画笔或图片资源，设置App标签内的BitsPerPixel属性，这里有更多信息。 - XAML页面设计视图：

ImageBrush

图片刷（Image Brush）使用一张图片绘制或者填充一个区域，指定它的ImageSource属性和Stretch属性（默认使用Fill）。 在XAML 中创建一个ImageBrush，请看以下代码片断。

```
<ImageBrush ImageSource="images/Flower.jpg" Stretch="Fill" AlignmentX="Center" AlignmentY="Center" />
```

在这里: AlignmentX: 用来设置水平方向对齐方式 AlignmentY: 用来设置垂直方向对齐方式

- PPImageBrush.xaml

```
<Grid x:Name="LayoutRoot">
        <controls:Panorama Title="image brush">
            <!--Panorama item one-->
            <controls:PanoramaItem Header="rectangle">
                <Grid>
                    <Rectangle Width="300" Height="250" >
                        <Rectangle.Fill>
                            <ImageBrush ImageSource="images/Flower.jpg"
                                        Stretch="Fill"
                                        AlignmentX="Center"
                                        AlignmentY="Center" />
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item two-->
            <controls:PanoramaItem Header="ellipse">
                <Grid>
                    <Ellipse Height="300" Width="300">
                        <Ellipse.Fill>
                            <ImageBrush ImageSource="images/Flower.jpg"
                                        AlignmentX="Center"
                                        AlignmentY="Center" />
                        </Ellipse.Fill>
                    </Ellipse>
                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item three-->
            <controls:PanoramaItem Header="text">
                <Grid>
                    <TextBlock Text="#WP7" FontSize="150" FontWeight="ExtraBlack" Margin="0,120">
                        <TextBlock.Foreground>
                            <ImageBrush ImageSource="images/Flower.jpg"
                                        Stretch="Fill"
                                        AlignmentX="Center"
                                        AlignmentY="Center" />
                        </TextBlock.Foreground>
                    </TextBlock>
                </Grid>
            </controls:PanoramaItem>
        </controls:Panorama>
    </Grid>
```


(c) Copyright Nokia 2013. All rights reserved.

- XAML页面设计视图:

VideoBrush

视频刷（Video Brush）使用MediaElement类的对象在指定区域绘制或填充运行中的视频。MediaElement对象支持音频和视频。在这个应用程序中，我使用mp4文件（视频文件）来播放视频，请看MSDN 上的Silverlight 文档来确认Windows Phone支持的媒体格式。除此之外，我也集成了live camera作为一个VideoBrush。在panorama的最后一项，你可以通过*camera open*按钮访问。 在XAML 中创建一个VideoBrush和MediaElement对象，请看以下代码片断。

```
<MediaElement x:Name="media1" Source="video/NokiaLumia710.mp4"
                              IsMuted="False"
                              Opacity="0.0"
                              Volume="10"
                              IsHitTestVisible="False"/>

<VideoBrush x:Name="viewVideo" SourceName="media1"/>
```

- PPVideoBrushes.xaml

```
<Grid x:Name="LayoutRoot">
        <MediaElement x:Name="media1"
                                  Source="video/NokiaLumia710.mp4"
                                  IsMuted="False"
                                  Opacity="0.0"
                                  Volume="10"
                                  IsHitTestVisible="False"  />

        <controls:Panorama x:Name="vp" Title="video brushes">

            <!--Panorama item one-->
            <controls:PanoramaItem Header="rectangle">
                <Grid>
                    <Rectangle Width="372" Height="340" Stroke="White" StrokeThickness="5" Margin="21,88,27,70">
                        <Rectangle.Fill>
                            <VideoBrush x:Name="viewVideo" SourceName="media1"/>
                        </Rectangle.Fill>
                    </Rectangle>
                </Grid>
            </controls:PanoramaItem>

            <controls:PanoramaItem Header="circle">
                <Grid>
                    <Ellipse Margin="44,98,48,106"
                        Height="294"
                        Width="328"
                        StrokeThickness="5"
                        Stroke="White">
                        <Ellipse.Fill>
                            <VideoBrush x:Name="viewVideoc" SourceName="media1"/>
                        </Ellipse.Fill>

                    </Ellipse>
                </Grid>
            </controls:PanoramaItem>

            <controls:PanoramaItem Header="polygon">
                <Grid>
                    <Polygon Points="5,270 270,270 270,10"
                             Stroke="White"
                             StrokeThickness="10"
                             Margin="51,104,57,114" Height="280" Width="312" >
                        <Polygon.Fill>
                            <VideoBrush x:Name="viewVideoP" SourceName="media1"/>
                        </Polygon.Fill>
                    </Polygon>
                </Grid>
            </controls:PanoramaItem>

            <controls:PanoramaItem Header="star">
                <Grid>
                    <Polygon Points="240 48, 352 396, 58 180, 422 180, 128 396"
                             Stroke="Magenta"
                             StrokeThickness="10" FillRule="NonZero" Margin="-22,0,2,0" Width="440">
                        <Polygon.Fill>
                            <VideoBrush x:Name="viewVideoT" SourceName="media1"/>
                        </Polygon.Fill>
```

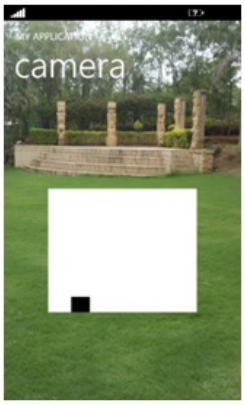

(c) Copyright Nokia 2013. All rights reserved.

```
                    </Polygon>

                </Grid>
            </controls:PanoramaItem>

            <!--Panorama item four-->
            <controls:PanoramaItem Header="text block">
                <Grid>
                    <TextBlock FontFamily="Verdana"
                               Margin="0,50"
                               FontSize="120"
                               FontWeight="Bold"
                               Text="PHONE">
                      <TextBlock.Foreground>
                         <VideoBrush SourceName="media1" />
                      </TextBlock.Foreground>
                    </TextBlock>
                </Grid>
            </controls:PanoramaItem>
            <!--Panorama item five-->
            <controls:PanoramaItem Header="camera">
                <Grid>
                    <Button Content="open camera" Height="72" HorizontalAlignment="Left" Margin="76,198,0,0" Name="button1" Vertical
                </Grid>
            </controls:PanoramaItem>
        </controls:Panorama>
    </Grid>
```

- XAML页面设计视图：

PPCamera.xaml 想要通过视频刷（video brush ）使用相机，点击相机按钮，它会切换至相机屏幕界面。在PPCamera.xaml.cs类中添加以下提到的代码。

```
public partial class PPCamera : PhoneApplicationPage
    {
        private PhotoCamera devicePhotoCamera;

        public PPCamera()
        {
            InitializeComponent();

            this.Loaded += new RoutedEventHandler(PPCamera_Loaded);
        }

        void PPCamera_Loaded(object sender, RoutedEventArgs e)
        {
            devicePhotoCamera = new PhotoCamera();
            viewCamera1.SetSource(devicePhotoCamera);
        }
    }
```



~ 快乐编程 ~

总结

Silverlight画笔API对于开发Windows Phone 7上Metro风格应用程序的优美UI (用户界面)非常有用。

(c) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.