GradBack

GradientBackground

By: Edward Willemsen, 2007

Table of Contents
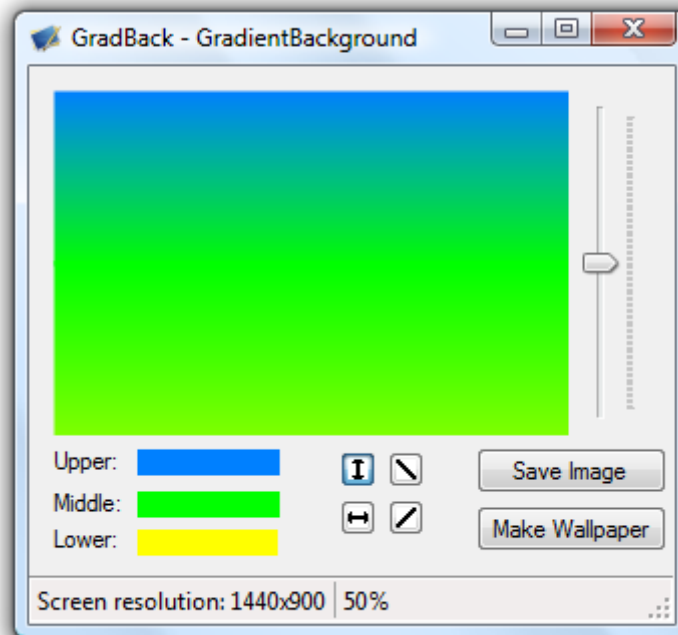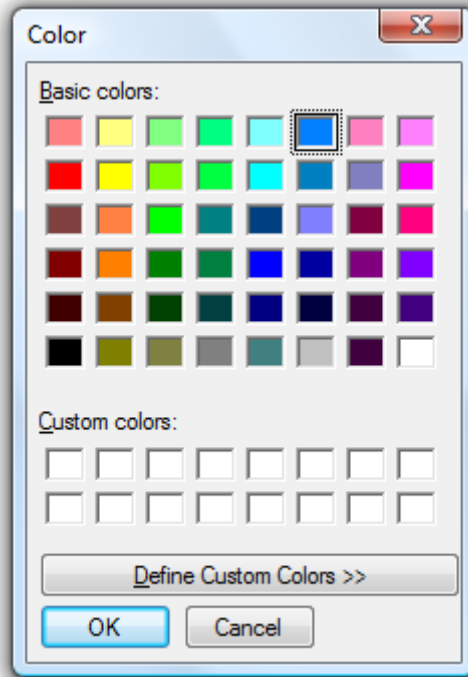
1. Common

Recently one of my colleagues asked my assistance to create small application which is able to make gradient wallpapers. The wallpaper preview can be used to get an idea of how the new wallpaper will look like. Next the wallpaper can be saved to a Windows Bitmap BMP file, a Joint Photographic Experts Group JPEG file or a Portable Network Graphics PNG file. And finally, using the Make Wallpaper button the in the preview shown gradient image can be set as wallpaper.

2. Program

When the program is started it will look like the following:

Within the program three color areas are shown called Upper, Middle and Lower. The lower color is only visible when a horizontal or vertical gradient is selected. It will disappear when one of the diagonal items is selected. The color of each of the colors can be change by clicking the color with the mouse.

On the right of the colors the four radio-bitmap-buttons are placed. By default the vertical gradient mode is selected. The following gradient patterns can be selected:

 Vertical Gradient Mode;

 Horizontal Gradient Mode;

 Backward Diagonal Gradient Mode;

 Forward Diagonal Gradient Mode.

2.1. Save Image

When the save image button is pushed the regular save file dialog will appear. GradBack is able to save the file in the following graphical file formats:

Extension	Description
BMP	Windows Bitmap
JPEG	Joint Photographic Experts Group
PNG	Portable Network Graphics

If you are a Microsoft Windows Vista user bear in mind that a known bug exists with regard to the common save dialog. The dialog will remind the last used file extension regardless of the change of the selected type. So when you select save as bitmap, it still creates a file with the JPG extension. This can only be avoided by adding the file extension in the dialog.

2.2. Make Wallpaper

The make wallpaper will change the current wallpaper similar to the one shown within the preview. The size of the wallpaper will be the same as the size of the primary desktop. This size is also shown within the status-strip.

2.3. Slider

The slider can be used in both vertical and horizontal gradient modes and changes the position of the middle color value. The percentage of the sliders position is shown within the status-strip.

3. TechoTalk

When you started using .NET v1.1 you're probably familiar with the ConfigurationManager. Well, within the .NET 2.0 Framework this has become obsolete. The following two snippets show the code I used to save and load the user's application settings.

3.1. Create Preferences

Within Visual Studio 2005 use the Project -> Program Properties item and select Settings. For this application the settings are shown here:

Name	Type	Scope	Value
ucolor	System.Dra...	User	RoyalBlue
mcolor	System.Dra...	User	Lime
lcolor	System.Dra...	User	Yellow
position	int	User	50
mode	System.Dra...	User	Vertical

3.2. Saving Preferences

We save the settings when the main form is closed.

```csharp
private void frmMain_FormClosed(object sender, FormClosedEventArgs e)
        {
            Properties.Settings.Default.ucolor = pnlU.BackColor;
            Properties.Settings.Default.mcolor = pnlM.BackColor;
            Properties.Settings.Default.lcolor = pnlL.BackColor;
            Properties.Settings.Default.position = tBar.Value;
            Properties.Settings.Default.mode = mode;
            Properties.Settings.Default.Save();
        }
```


3.3. Loading Preferences

We load the settings when the main form is shown.

```csharp
private void frmMain_Shown(object sender, EventArgs e)
        {
            pnlU.BackColor = Properties.Settings.Default.ucolor;
            pnlM.BackColor = Properties.Settings.Default.mcolor;
```

```
        pnlL.BackColor = Properties.Settings.Default.lcolor;
        mode = Properties.Settings.Default.mode;
        tBar.Value = Properties.Settings.Default.position;

        switch (mode)
        {
            case LinearGradientMode.Vertical:
              rbV.Checked=true;
              break;
            case LinearGradientMode.Horizontal:
              rbH.Checked = true;
              break;
            case LinearGradientMode.ForwardDiagonal:
              rbFD.Checked = true;
              break;
            case LinearGradientMode.BackwardDiagonal:
              rbBD.Checked = true;
              break;
        }

        sbLbl.Text = "Screen resolution: " +
              SystemInformation.PrimaryMonitorSize.Width + "x" +
              SystemInformation.PrimaryMonitorSize.Height;
        sbPos.Text = tBar.Value.ToString() + "%";
    }
```

By using the switch we select the gradient mode how it was left by the user the last time the program was used or by using the application default value.

3.4. Making the Wallpaper

To be able to create a wallpaper we have to use unmanaged code.

```
[DllImport("user32.dll", CharSet = CharSet.Auto)]
static extern int SystemParametersInfo(int uAction, int uParam,
                                    string lpvParam, int fuWinIni);

const int SPI_SETDESKWALLPAPER    = 20;
const int SPIF_UPDATEINIFILE      = 0x01;
const int SPIF_SENDWININICHANGE   = 0x02;

SystemParametersInfo(SPI_SETDESKWALLPAPER, 0, filepath,

      SPIF_UPDATEINIFILE | SPIF_SENDWININICHANGE);
```

Within the program we use the bitmap format as the wallpaper. When you are a Microsoft Windows Vista user you can also use JPEG or PNG for the wallpaper, but users with older operating systems can only use BMPs.

4. History

Version	Comment
1.0	First release of the application.