

Forum:[Hack Forums](#)

Thread: [Viewing two live streams at once](#)

shinigami_sam - January 1, 2015, 6:19 am

I'm trying to stream two live feeds from two different sources using Microsoft expression encoder 4 on WPF platform, i tried to streaming one feed only and it works fine and i can switch between the two sources, but when i try to stream both of them on different panels it doesn't show anything.

I don't know what i'm doing wrong here as i copied and pasted the code for 1 stream and made it for two as shown in the codes i provided.

XAML for the two streams :

```
Code <Window x:Class="WpfApplication1.FullscreenMode"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:wf="clr-namespace:System.Windows.Forms;assembly=System.Windows.Forms"

    xmlns:WPFMediaKit="clr-namespace:WPFMediaKit.DirectShow.Controls;assembly=WPFMediaKi
t"
    x:Name="MainWindow" Title="FullscreenMode" Height="768" Width="1366"
    WindowStartupLocation="CenterScreen" WindowStyle="None" AllowsTransparency="True"
    WindowState="Maximized" Loaded="MainWindow_Loaded" >
    <Window.Background>
        <SolidColorBrush Opacity="0.86" Color="Black"/>
    </Window.Background>
    <Grid>

        <UserControl xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
            xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
            xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
            xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
            xmlns:wf="clr-namespace:System.Windows.Forms;assembly=System.Windows.Forms"
            MinHeight="480" MinWidth="640"
            x:Name="livevideo1"
            VerticalAlignment="Center" HorizontalAlignment="Center" Grid.Column="0"
            Margin="20,42,694,211" Visibility="Visible" mc:Ignorable="d" Height="484" Width="644" >
                <Border x:Name="RecordingBorder1" BorderThickness="2" BorderBrush="Black">
                    <WindowsFormsHost x:Name="WinFormsHost1" Margin="0,0,0,0" Background="{x:Null}"
                    Width="640" Height="480">
                        <wf:Panel x:Name="viewpanel1"/>
                    </WindowsFormsHost>
                </Border>
            </UserControl>

        <UserControl xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
            xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
            xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
            xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
```

```

xmlns:wf="clr-namespace:System.Windows.Forms;assembly=System.Windows.Forms"
    MinHeight="480" MinWidth="640"
    x:Name="livevideo2"
    VerticalAlignment="Center" HorizontalAlignment="Center" Grid.Column="0"
Margin="692,42,22,211" Visibility="Visible" mc:Ignorable="d" Height="484" Width="644" >
    <Border x:Name="RecordingBorder2" BorderThickness="2" BorderBrush="Black">
        <WindowsFormsHost x:Name="WinFormsHost2" Margin="0,0,0,0" Background="{x:Null}"
Width="640" Height="480">
            <wf:Panel x:Name="viewpanel2"/>
        </WindowsFormsHost>
    </Border>
</UserControl>

    <Label Content="Video source 1" Margin="210,572,0,0" Width="88" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="32" Foreground="White"/>
    <ComboBox x:Name="AvailableVideoSources1" VerticalAlignment="Top"
HorizontalAlignment="Left" Width="150" Height="25" Margin="300,573,0,0"
SelectionChanged="AvailableVideoSources1_SelectionChanged"/>
    <Label Content="Video source 2" Margin="891,572,0,0" Width="88" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Foreground="White"/>
    <ComboBox x:Name="AvailableVideoSources2" VerticalAlignment="Top"
HorizontalAlignment="Left" Width="150" Height="25" Margin="985,572,0,0"
SelectionChanged="AvailableVideoSources2_SelectionChanged"/>

```

</Grid>

Code Behind:

Code public partial class FullscreenMode : Window

```

{
    /// <summary>
    /// Creates job for capture of live source
    /// </summary>
    private LiveJob _job1, _job2;

    /// <summary>
    /// Device for live source
    /// </summary>
    private LiveDeviceSource _deviceSource1, _deviceSource2;

    EncoderDevice Vdevice1, Adevice1, Vdevice2, Adevice2 = null;

    string VideoSource1= "", VideoSource2= "";

    private void ActivateVideosource1()
    {
        // Starts new job for preview window
        _job1 = new LiveJob();

        _deviceSource1 = null;

        if (_deviceSource1 == null)
        {
            // Create a new device source. We use the first audio and video devices on the system

```

```

        _deviceSource1 = _job1.AddDeviceSource(Vdevice1, Adevice1);
        _deviceSource1.PickBestVideoFormat(new System.Drawing.Size(640, 480), 10000000 /
30);
    }

    // Get the properties of the device video
    SourceProperties sp1 = _deviceSource1.SourcePropertiesSnapshot();

    // Resize the preview panel to match the video device resolution set
    viewpanel1.Size = new System.Drawing.Size(sp1.Size.Width, sp1.Size.Height);

    // Setup the output video resolution file as the preview
    _job1.OutputFormat.VideoProfile.Size = new System.Drawing.Size(sp1.Size.Width,
sp1.Size.Height); ;

    _job1.OutputFormat.VideoProfile.Bitrate = new ConstantBitrate(1000);

    // Sets preview window to winform panel hosted by xaml window
    _deviceSource1.PreviewWindow = new PreviewWindow(new HandleRef(viewpanel1,
viewpanel1.Handle));

    // Make this source the active one
    _job1.ActivateSource(_deviceSource1);
}

private void ActivateVideosource2()
{
    // Starts new job for preview window
    _job2 = new LiveJob();

    _deviceSource2 = null;

    if (_deviceSource2 == null)
    {
        // Create a new device source. We use the first audio and video devices on the system
        _deviceSource2 = _job2.AddDeviceSource(Vdevice2, Adevice2);
        _deviceSource2.PickBestVideoFormat(new System.Drawing.Size(640, 480), 10000000 /
30);
    }

    // Get the properties of the device video
    SourceProperties sp2 = _deviceSource2.SourcePropertiesSnapshot();

    // Resize the preview panel to match the video device resolution set
    viewpanel2.Size = new System.Drawing.Size(sp2.Size.Width, sp2.Size.Height);

    // Setup the output video resolution file as the preview
    _job2.OutputFormat.VideoProfile.Size = new System.Drawing.Size(sp2.Size.Width,
sp2.Size.Height); ;

    _job2.OutputFormat.VideoProfile.Bitrate = new ConstantBitrate(1000);

    // Sets preview window to winform panel hosted by xaml window

```

```

        _deviceSource2.PreviewWindow = new PreviewWindow(new HandleRef(viewpanel2,
viewpanel2.Handle));

        // Make this source the active one
        _job2.ActivateSource(_deviceSource2);
    }

    public FullscreenMode()
    {
        InitializeComponent();

        foreach (EncoderDevice edv in EncoderDevices.FindDevices(EncoderDeviceType.Video))
        {
            AvailableVideoSources1.Items.Add(edv.Name.ToString());
        }

        foreach (EncoderDevice edv in EncoderDevices.FindDevices(EncoderDeviceType.Video))
        {
            AvailableVideoSources2.Items.Add(edv.Name.ToString());
        }
    }

    private void MainWindow_Loaded(object sender, RoutedEventArgs e)
    {
        foreach (EncoderDevice eda in EncoderDevices.FindDevices(EncoderDeviceType.Audio))
        {
            Adevice1 = eda;
        }

    }

    private void AvailableVideoSources1_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {
        VideoSource1 = AvailableVideoSources1.SelectedValue.ToString();
        if (VideoSource1 != VideoSource2)
        {
            foreach (EncoderDevice edv in EncoderDevices.FindDevices(EncoderDeviceType.Video))
            {
                if (edv.Name == VideoSource1)
                {
                    Vdevice1 = edv;
                }
            }

            ActivateVideosource1();
        }
    }

    private void AvailableVideoSources2_SelectionChanged(object sender,

```

SelectionChangedEventArgs e)

```
{
    VideoSource2 = AvailableVideoSources2.SelectedValue.ToString();
    if (VideoSource2 != VideoSource1)
    {
        foreach (EncoderDevice edv in EncoderDevices.FindDevices(EncoderDeviceType.Video))
        {
            if (edv.Name == VideoSource2)
            {
                Vdevice2 = edv;
            }
        }

        ActivateVideosource2();
    }
}
```

The selection changed events trigger and everything works fine - code wise - but i see no picture on the panels.

Znooteh - January 1, 2015, 12:27 pm

I'm pretty(very) new to C#, but I would make one thread for each video if that's not what you're doing

shinigami_sam - January 1, 2015, 1:11 pm

(Jan 1, 2015 01:27 PM)Motherflufferr Wrote: [\[CITE\]](#) I'm pretty(very) new to C#, but I would make one thread for each video if that's not what you're doing

1st of all thanks for replying, 2nd i figured it out it was the AllowsTransparency="True" property on the main window that cause all that - i don't know why tho - but i removed it and it's working the way it's supposed to be now.