How to create a toggle image on Windows Phone

This article explains how to use visual states to animate collapsing and expanding an image on the page.

08 Dec 2013

Introduction

A picture is worth a thousand words but, even so, sometimes it can take too much screen real estate. There are a number of approaches to this problem: one is to display only part of the image and provide a mechanism so it can be expanded to full size.

This article shows how to create a page containing both text and an image such that the page opens with the image only partially visible - in order to draw attention to the text. The image can be expanded and collapsed when tapped by the user. This is accomplished using visual states and a nice animation.

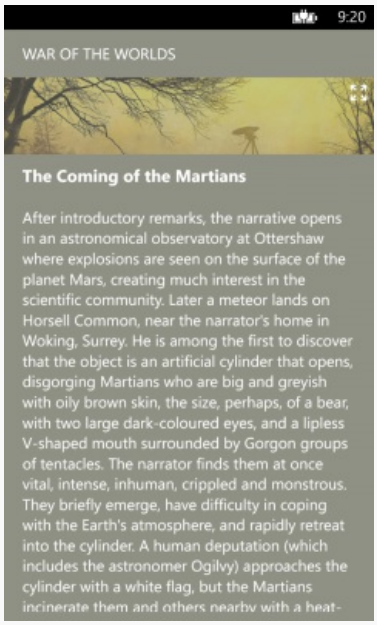


Image closed

Image opened

Implementation

Create the initial page

First set up the initial state of the page, create a new basic Windows Phone project and add following XAML to the `ContentPanel` of the **MainPage.xaml**:

```
<Grid x:Name="ContentPanel" Grid.Row="1" Margin="0">
    <Grid.RowDefinitions>
        <RowDefinition Height="Auto" />
        <RowDefinition Height="*" />
    </Grid.RowDefinitions>

    <Image x:Name="Image" HorizontalAlignment="Left" VerticalAlignment="Top"
   Grid.Row="0"
   Source="/Assets/war_of_the_worlds_doomsday_final.jpg" Opacity="0.4" Height="100"
Width="480" Stretch="UniformToFill" Tap="image_Tap"/>
    <Image x:Name="IconExpand" Grid.Row="0" Width="40"
            Source="/Assets/arrow.expand.png" HorizontalAlignment="Right"
VerticalAlignment="Top" IsHitTestVisible="False" />
    <ScrollViewer Name="ScrollViewer"
                  Grid.Row="1"
```


(C) Copyright Nokia 2014. All rights reserved.

```
                    Margin="0,12,0,12"
                    VerticalAlignment="Top"
                    VerticalScrollBarVisibility="Visible"
  >
        <RichTextBox Name="Article" Margin="12,0,12,0">
            <Paragraph>
                <Run FontWeight="Bold" FontSize="22" Text="The Coming of the Martians"
/>
            </Paragraph>
            <Paragraph>
                <LineBreak />
                <Run Text="After introductory remarks, the narrative opens in an
astronomical observatory at Ottershaw where explosions are seen on the surface of the
planet Mars, creating much interest in the scientific community. Later a meteor lands on
Horsell Common, near the narrator's home in Woking, Surrey. He is among the first to
discover that the object is an artificial cylinder that opens, disgorging Martians who
are big and greyish with oily brown skin, the size, perhaps, of a bear, with two large
dark-coloured eyes, and a lipless V-shaped mouth surrounded by Gorgon groups of
tentacles. The narrator finds them at once vital, intense, inhuman, crippled and
monstrous. They briefly emerge, have difficulty in coping with the Earth's atmosphere,
and rapidly retreat into the cylinder. A human deputation (which includes the astronomer
Ogilvy) approaches the cylinder with a white flag, but the Martians incinerate them and
others nearby with a heat-ray before beginning to assemble their machinery. Military
forces arrive that night to surround the common, including Maxim guns. The population of
Woking and the surrounding villages are reassured by the presence of the military. A
tense day begins, with much anticipation of military action by the narrator." />
            </Paragraph>
        </RichTextBox>
    </ScrollViewer>
</Grid>
```

With that in place you get a page that resembles the first screenshot above. The image is not fully visible and we've tweaked the opacity to draw less attention to it. There is also a small icon in the top right hand corner to provide a hint to the user that the image can be enlarged.

Create the open and closed states

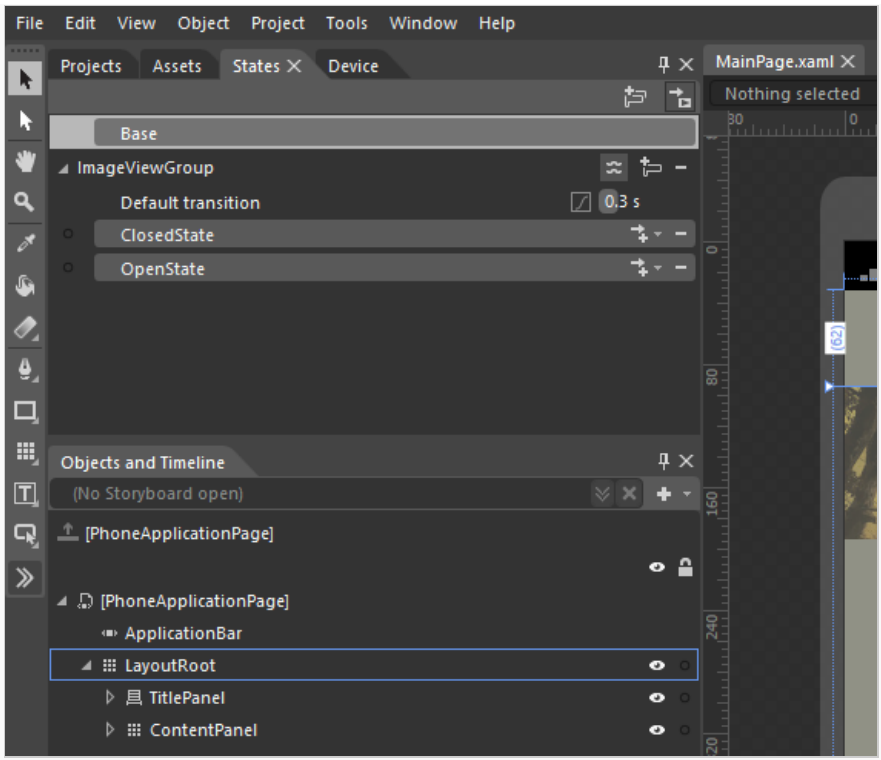
For things like this I tend to use *Blend*, because it offers a visual interface to define and work with **Visual States**!

1. Open the project in *Blend* and go to the *States* tab, click on the **Add state group** icon and name your new group **ImageViewGroup**
2. Select the newly added state group and click twice on the **Add state** icon, name those states **ClosedState** and **OpenState**.

Also be sure to select the **Default transition** and click on the **EasingFunction** icon - this will set the actual visual animation that will be used when switching from one visual state to the other. I've selected the **Cubic InOut**, but be sure to try out a few that fits your taste! Next, we need to define how long this animation will take. So to the right of the **EasingFunction** icon, set the amount of seconds you prefer. I've chosen 0.3 seconds, but this also is something you need to test out.

Check the Visual states screenshot to verify your settings

(C) Copyright Nokia 2014. All rights reserved.

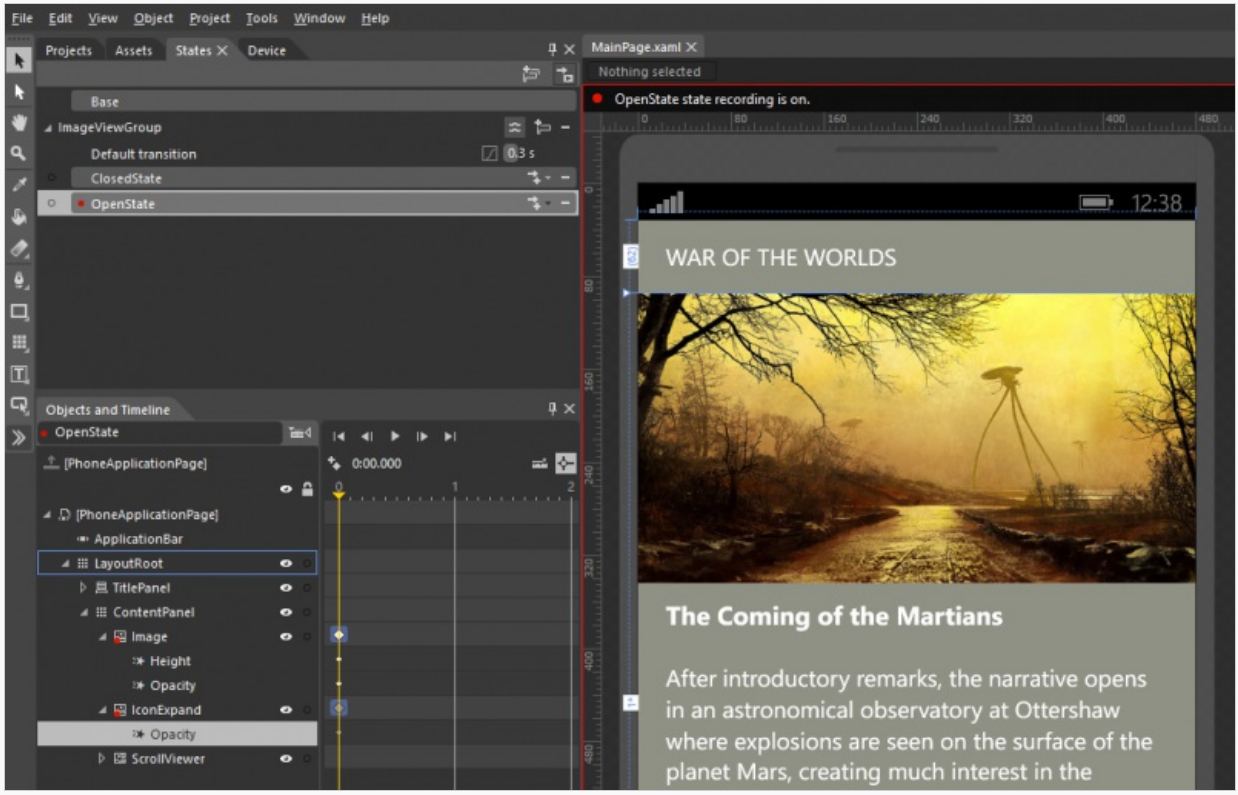


Visual states

Now select the **OpenState** line an be sure to notice that *Blend* tells you that **OpenState state recording is on** in the middle on the **MainPage.xaml** tab. In the **Objects and Timeline** tab, open up the **ContentPanel** and select the **Image**. When the **Image** is selected, change it's **Height** to 250 in the *Properties* tab on the right, also change the **Opacity** to 100%. Click on the **IconExpand** and set it's **Opacity** to 0%

Note: We are tweaking the opacity on the `IconExpand` and not the visibility, because the opacity will have a smoother transition

Check the OpenState settings screenshot to verify your settings



OpenState settings

For the **ClosedState** we do something similar, so select it in *Blend* and again open the **ContentPanel**. Select the **Image** and set it's **Opacity** back to the original value 40%. That's all for this state, so you can now save everything and close *Blend*.

(C) Copyright Nokia 2014. All rights reserved.

Detect tap and open animation

There is only one more thing left to do and that is trigger the change from one state to the other, when the user taps the image! So open the project again in Visual Studio and open the code behind of the **MainPage.xaml**. In here add the following code underneath the constructor:

```
private bool _isOpen = false;
private void image_Tap(object sender, System.Windows.Input.GestureEventArgs e)
{
 if (_isOpen)
 {
 VisualStateManager.GoToState(this, "ClosedState", true);
 _isOpen = false;
 }
 else
 {
 VisualStateManager.GoToState(this, "OpenState", true);
 _isOpen = true;
 }
}
```

We are using the `VisualStateManager` that is build in the Windows Phone SDK to go to the correct state when the user taps the image. We also need to keep track what state we are in to go to the other one.

With this all in place you get a nice visual animation when the user taps the image, to open or close it.

Video demo

The media player is loading...

Downloads

A complete working demo can be downloaded here: File:ImageSelection.zip

(C) Copyright Nokia 2014. All rights reserved.