Page transition in Qt and WP7

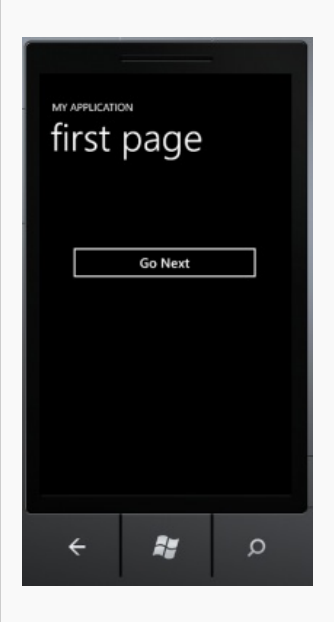
This article demonstrates how to create page navigation with sliding transition effect in Qt and WP7

Introduction

In this article we will see how to navigate page using sliding transition. Though in Qt and WP7 we have used different technique to bring the same kind of effect. In WP7 we have used Silverlight Toolkit to navigate the complete page and for Qt we used State and Transition to navigate elements inside a ListView. But in both the cases we can have almost the same feel.

We create two pages, when user clicks on the first page's button it takes him to the second page with a sliding transition view. And the user can also come back to the first page when he clicks on the back button.

Sliding Transition in Qt



Sliding Transition in WP7

Note: The transition effect is not clear in these images, to get the actual effect please download the source code

Implementation

Let's create an empty project for both Qt and WP7. First we will work on Qt project and then will move on to WP7 project.

Qt Project (main.qml)

Both the First and the Second pages are Items which have a ListView and Button . When user clicks on the button the default State of the object get changed and PropertyChanges enables the Items's property value to its current state to new state. This creates a sliding effect in page navigation.

```
//Page 1
        Item {
            id: view1
            width: parent.width
            height: parent.height
           ListView {
               Text {
                   id: header1
                   text: qsTr("First Page")
                   color: "white"
                   font.pixelSize: 30
                   font.bold: true;
                   anchors.horizontalCenter: parent.horizontalCenter;
                   anchors.top :parent.top
               }
               Button{
               id: button1
               text: "Go next"
               anchors.verticalCenter: parent.verticalCenter
               anchors.horizontalCenter: parent.horizontalCenter
               onClicked:
               {
                   loadSecond();
               }
              }
                id: lView1;
```


(C) Copyright Nokia 2013. All rights reserved.

```
                width: parent.width; height: parent.height; x: -(parent.width * 1.5); cacheBuffer: 100;
            }
            states:
                State {
                name: "ListView"; when: screen.inFirstScreen == true
                PropertyChanges { target: lView1; x: 0 }
                PropertyChanges { target: photoListViewIngredient2; x: -(parent.width * 1.5) }
                        }
            transitions: Transition {
                NumberAnimation { properties: "x"; duration: 700; easing.type: Easing.InOutQuad }
            }
    }


        //Page 2
        Item {
            id: view2
            width: parent.width
            height: parent.height
            anchors.top: parent.top; anchors.bottom: parent.bottom
           ListView {
               Text {
                   id: header2
                   text: qsTr("Second Page")
                   color: "white"
                   font.pixelSize: 30
                   font.bold: true;
                   anchors.horizontalCenter: parent.horizontalCenter;
                   anchors.top :parent.top
               }
               Image {
                   id: image2
                   source: "Nokia-Lumia-800-Logo.jpg"
                   anchors.top:header2.bottom
                   anchors.horizontalCenter: parent.horizontalCenter
                   height: 170
                   width: 320
               }
               Button{
               id: button2
               text: "Go back"
               anchors.verticalCenter: parent.verticalCenter
               anchors.horizontalCenter: parent.horizontalCenter
               onClicked:
               {
                   loadFirst();
               }
              }
                id: lView2;
                width: parent.width; height: parent.height; x: -(parent.width * 1.5); cacheBuffer: 100;
            }
            states:
                State {
                name: "ListView"; when: screen.inSecondScreen == true
                PropertyChanges { target: lView2; x: 0 }
                PropertyChanges { target: lView1; x: -(parent.width * 1.5) }
                        }
            transitions: Transition {
                NumberAnimation { properties: "x"; duration: 700; easing.type: Easing.InOutQuad }
            }
         }
```

WP7 Project (MainPage.xaml)

Let's first download theSilverlight Toolkit and add the reference `Microsoft.Phone.Controls.Toolkit` to the project. We will add a new XAML file (*NewPage.xaml*), this page will play the same role as that of the SecondPage, where as *MainPage.xaml* plays the same role as that of FirstPage in Qt . Add the namespace reference in both the XAML files

```
xmlns:toolkit="clr-namespace:Microsoft.Phone.Controls;assembly=Microsoft.Phone.Controls.Toolkit"
```

Now let's add the transition code in both the XAML file, so that when user navigates each page it give a sliding effect .

```
<toolkit:TransitionService.NavigationInTransition>
        <toolkit:NavigationInTransition>
            <toolkit:NavigationInTransition.Backward>
                <toolkit:SlideTransition Mode="SlideRightFadeIn"/>
            </toolkit:NavigationInTransition.Backward>
            <toolkit:NavigationInTransition.Forward>
                <toolkit:SlideTransition Mode="SlideRightFadeIn"/>
            </toolkit:NavigationInTransition.Forward>
        </toolkit:NavigationInTransition>
    </toolkit:TransitionService.NavigationInTransition>
    <toolkit:TransitionService.NavigationOutTransition>
        <toolkit:NavigationOutTransition>
            <toolkit:NavigationOutTransition.Backward>
                <toolkit:SlideTransition Mode="SlideLeftFadeOut"/>
            </toolkit:NavigationOutTransition.Backward>
            <toolkit:NavigationOutTransition.Forward>
                <toolkit:SlideTransition Mode="SlideLeftFadeOut"/>
            </toolkit:NavigationOutTransition.Forward>
        </toolkit:NavigationOutTransition>
    </toolkit:TransitionService.NavigationOutTransition>
```

At this stage if we compile and run the application we will see that there is no transition while page navigation. To get the transition during page navigation, edit the method `InitializePhoneApplication()` of *App.xaml.cs* file. Replace the line

```
RootFrame = new PhoneApplicationFrame();
```

To

```
RootFrame = new TransitionFrame();
```

The Sliverlight Toolkit also provides other types of transition styles like Turnstile, Swivel, Rotate, etc.

(C) Copyright Nokia 2013. All rights reserved.

Source Code

- The full source code of Qt example is available here: File:PageTransitionQt.zip
- The full source code of WP7 example is available here: File:PageTransitionWP7.zip

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.