Earthquake-Reporting application for Qt and WP7

Introduction

This article demonstrates how to implement the Qt Quick Earthquake-Reporting application with XmlListModel application on Windows Phone 7. The data is obtained from the U.S. Geological Survey and renders the result as a list of earthquake events using a ListBox.

Earthquake Reporting Application

Implementation

Qt

The article Earthquake-Reporting application with XmlListModel explains how the app is implemented in Qt Quick using the XmlListModel QML Element that lets you fetch and parse RSS feeds using only `XPath` queries

WP7

First create an empty Windows Phone project. We add a ListBox to display the data to the left of the screen.

```
<StackPanel Orientation="Horizontal"  HorizontalAlignment="Left" VerticalAlignment="Top"  Width="350" >
                    <ListBox Name ="listboxData" Margin="0,0,-12,0"  Loaded="Menu_Loaded"
                  SelectionChanged="listboxData_SelectionChanged" >
                <ListBox.ItemTemplate>
                    <DataTemplate>
                        <StackPanel Orientation="Horizontal" Margin="0,0,0,17">
                                 <StackPanel Width="350" Margin="12,-9,0,0">
                                    <TextBlock Text="{Binding Title}" TextWrapping="Wrap" Style="{StaticResource PhoneTextLargeStyle
                                <TextBlock Text="{Binding PubDate}" TextWrapping="Wrap" Margin="11,-6,12,0"  Style="{StaticResource
                            </StackPanel>

                        </StackPanel>
                    </DataTemplate>
                </ListBox.ItemTemplate>
            </ListBox>

            </StackPanel>
```

We download the data, parse it and bind it to the ListBox.

```
XElement xmlData = XElement.Parse(e.Result);
                    XNamespace geo = "http://www.w3.org/2003/01/geo/wgs84_pos#";
                    XNamespace dc = "http://purl.org/dc/elements/1.1/";
                    listboxData.Items.Clear();

                    foreach (var item in xmlData.Descendants("item"))
                    {

                        RssData rssData = new RssData();
                        rssData.Title = (string)item.Element("title").Value;
                        rssData.PubDate = (string)item.Element("pubDate").Value;
                        rssData.lat = (string)item.Element(geo + "lat").Value;
                        rssData.lng = (string)item.Element(geo + "long").Value;

                        listboxData.Items.Add(rssData);
                     }
```

When user clicks on any of the list items we display the detail information to the right of the screen.

```
<Canvas Width="350" Height="313" VerticalAlignment="Top" HorizontalAlignment="Right">
                <TextBlock Canvas.Left="10" Canvas.Top="141" Height="30" Name="textBlock1" Text="Lat : " />
                <TextBlock Canvas.Left="86" Canvas.Top="141" Height="30" Name="canLat" Text=""  TextWrapping="Wrap" />
                <TextBlock Canvas.Left="10" Canvas.Top="201" Height="30" Name="textBlock3" Text="Long : " />
                <TextBlock Canvas.Left="86" Canvas.Top="201" Height="30" Name="canLng" TextWrapping="Wrap" Text="" />
                <TextBlock Canvas.Left="10" Canvas.Top="21" Height="66" Name="canTitle" Text="" Width="323" TextWrapping="Wrap" />
                <TextBlock Canvas.Left="10" Canvas.Top="89" Height="40" Name="canSubTitle" Text="" Width="323" TextWrapping="Wrap" /
</Canvas>
```


(C) Copyright Nokia 2013. All rights reserved.

That's it.

Source Code

- The full source code of the example is available here: File:EarthQuakesWP7.zip

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.