向下隧道传播的路由事件。

例：程序routed\_event\_test2

MainWindow.xaml

<Window x:Class="routed\_event\_test1.MainWindow"

x:Name="Window"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:local="clr-namespace:routed\_event\_test1"

Title="MainWindow" Height="350" Width="255"

local:ReportLocationButton.ReportTime="ReportTimeHandler">

<Grid x:Name="gridRoot" Background="Lime" local:ReportLocationButton.ReportTime="ReportTimeHandler">

<Grid x:Name="gridA" Margin="10" Background="Blue"

local:ReportLocationButton.ReportTime="ReportTimeHandler">

<Grid.ColumnDefinitions>

<ColumnDefinition />

<ColumnDefinition />

</Grid.ColumnDefinitions>

<Canvas x:Name="canvasLeft" Grid.Column="0" Background="Red" Margin="10,10,10,165"

local:ReportLocationButton.ReportTime="ReportTimeHandler">

<local:ReportLocationButton x:Name="buttonLeft" Width="65" Height="100" Margin="10" Content="Left"

local:ReportLocationButton.ReportTime="ReportTimeHandler"/>

</Canvas>

<Canvas x:Name="canvasRight" Grid.Column="1" Background="Yellow" Margin="10,10,10,165"

local:ReportLocationButton.ReportTime="ReportTimeHandler">

<local:ReportLocationButton x:Name="buttonRight" Width="65" Height="100" Margin="10" Content="Right"

local:ReportLocationButton.ReportTime="ReportTimeHandler"/>

</Canvas>

<ListBox x:Name="listbox" Grid.ColumnSpan="2" HorizontalAlignment="Left"

Height="134" Margin="10,165,0,0" VerticalAlignment="Top" Width="207"/>

</Grid>

</Grid>

</Window>

ReportLocationButton.cs

// Copyright 2016.刘珅珅

// author：刘珅珅

// 路由事件

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*;

namespace routed\_event\_test1

{

// 路由事件参数

public class ReportedLocationEventArgs:*RoutedEventArgs*

{

public ReportedLocationEventArgs(*RoutedEvent* routedEvent, object source)

: base(routedEvent, source)

{ }

public string Location { get; set; }

}

public class ReportLocationButton:*Button*

{

// 声明并定于路由事件

// 向下隧道的路由事件

public static readonly *RoutedEvent* ReportLocationEvent = *EventManager*

.*RegisterRoutedEvent*("ReportTime", *RoutingStrategy*.*Tunnel*,

typeof(*EventHandler*<ReportedLocationEventArgs>), typeof(ReportLocationButton));

// CLR事件包装器

public event *RoutedEventHandler* ReportTime

{

add { *AddHandler*(ReportLocationEvent, value); }

remove { *RemoveHandler*(ReportLocationEvent, value); }

}

// 激发路由事件

protected override void OnClick()

{

base.OnClick();

ReportedLocationEventArgs args = new ReportedLocationEventArgs(ReportLocationEvent, this);

args.Location = this.*Name*;

this.*RaiseEvent*(args); // 激发路由事件

}

}

}

MainWindow.xaml.cs

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

using *System*.*Windows*;

using *System*.*Windows*.*Controls*;

using *System*.*Windows*.*Data*;

using *System*.*Windows*.*Documents*;

using *System*.*Windows*.*Input*;

using *System*.*Windows*.*Media*;

using *System*.*Windows*.*Media*.*Imaging*;

using *System*.*Windows*.*Navigation*;

using *System*.*Windows*.*Shapes*;

namespace routed\_event\_test1

{

/// <summary>

/// MainWindow.xaml 的交互逻辑

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

*InitializeComponent*();

}

// 路由事件的事件处理函数

private void ReportTimeHandler(object sender, ReportedLocationEventArgs e)

{

e.Location = (sender as *FrameworkElement*).*Name*;

string l = e.Location;

string c = "我到达了: " + l;

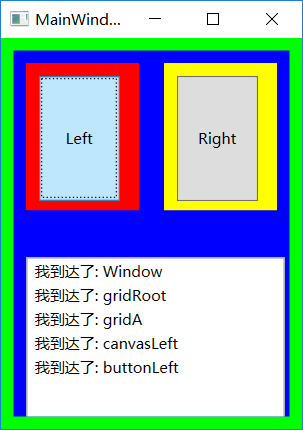
this.listbox.*Items*.*Add*(c);

}

}

}

输出结果



从结果可以看出，向下隧道传播时，事件由外向内传播。