

Zdarzenia krążące - WPF .NET

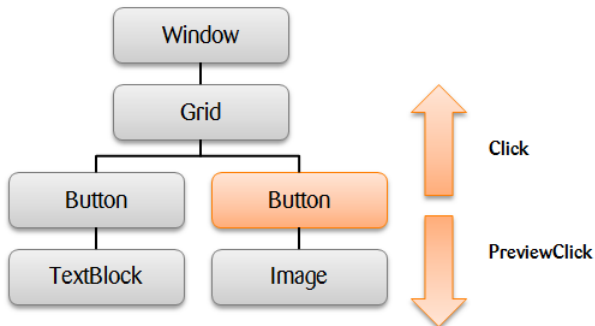
Dawid Adamski

Uniwersytet Mikołaja Kopernika w Toruniu

Co to właściwie jest?

Zdarzenie, które nie jest tylko wywoływane na obiekcie, którego dotyczy, ale na całym drzewie (zarówno logicznym jak i wizualnym).

Jakie drzewo ?



Rodzaje zdarzeń krążących

- Direct (bezpośrednie)
- Bubbling (propagacji)
- Tunneling (tunelowania)

Bezpośredni - Direct

Zdarzenie zgłaszane na elemencie źródłowym.

Tunelowania - Tunneling

Najpierw wywołany na elemencie głównym, a następnie przesuwa się w kierunku źródła. Chyba, że zostanie wcześniej obsłużony.

Propagacji - Bubbling

Najpierw wywołany na elemencie źródłowym, a następnie przesuwana się w kierunku rodziców. Oczywiście chyba, że zostanie wcześniej obsłużony.

Tworzenie własnego zdarzenia krążącego

```
// Rejestrujemy zdarzenie krążące
public static readonly RoutedEvent SelectedEvent =
   EventManager.RegisterRoutedEvent( "Selected", RoutingStrategy.Bubble,
    typeof(RoutedEventHandler), typeof(MyCustomControl));

// .NET wrapper (opakowanie ;) )
public event RoutedEventHandler Selected
{
    add { AddHandler(SelectedEvent, value); }
    remove { RemoveHandler(SelectedEvent, value); }
}

// Postawienie
RaiseEvent(new RoutedEventArgs(MyCustomControl.SelectedEvent));
```


RoutedEventArgs

Klasa, o której warto wspomnieć.

- Source - identyfikuje obiekt wywołujący zdarzenie.
- OriginalSource - identyfikuje obiekt, który jako pierwszy wywołał zdarzenie.
- Handled - pozwala na zatrzymanie Propagacji lub Tunelowania.
- RoutedEvent - identyfikuje, które zdarzenie właśnie krąży.

Źródła

<http://msdn.microsoft.com/en-us/library/ms742806.aspx>

<http://www.wpftutorial.net/RoutedEvents.html>

<http://msdn.microsoft.com/en-us/library/ms752288.aspx>

<http://patryknet.blogspot.com/2010/07/dlaczego-warto-uzywac-dependency.html>

<http://msdn.microsoft.com/en-us/library/ms742550.aspx>

Dziękuję za uwagę!

A teraz możemy coś popisać :)