

# C# Essentials WPF controls

Koen Bloemen



**DE HOGESCHOOL  
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt, [www.pxl.be](http://www.pxl.be)





- Panels

# Panels

- Eén van de belangrijkste soorten WPF controls
- Fungeren als containers voor andere besturingselementen
- Bepalen de lay-out van uw vensters
- Soorten panels:
  - Grid
  - Canvas
  - StackPanel
  - WrapPanel
  - DockPanel
  - Viewbox

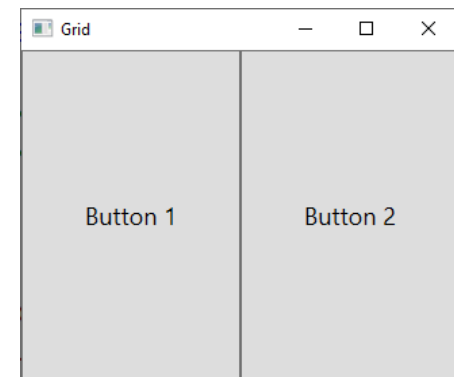


# Grid

- Een grid kan meerdere rijen en kolommen bevatten
- Voor de eerste component (vb. Button) moet je niet vertellen welke kolom of rij deze is. Maar het mag wel.
  - `Grid.Column="0" Grid.Row="0"`

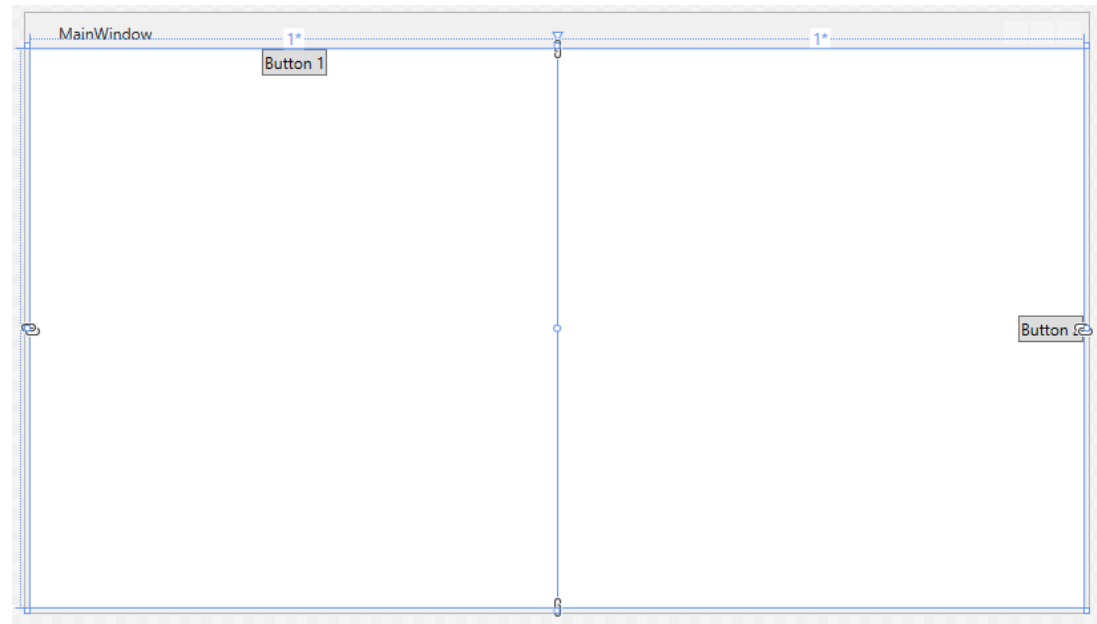
```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="*" />
  </Grid.ColumnDefinitions>
  <Button Content="Button 1" />
  <Button Grid.Column="1" Content="Button 2" />
</Grid>
```

Nemen beide evenveel kolommen of breedte in



# Grid

```
<Grid>  
  <Grid.ColumnDefinitions>  
    <ColumnDefinition Width="*" />  
    <ColumnDefinition Width="*" />  
  </Grid.ColumnDefinitions>  
  <Button VerticalAlignment="Top" HorizontalAlignment="Center">Button 1</Button>  
  <Button Grid.Column="1" VerticalAlignment="Center" HorizontalAlignment="Right">  
    Button 2  
  </Button>  
</Grid>
```

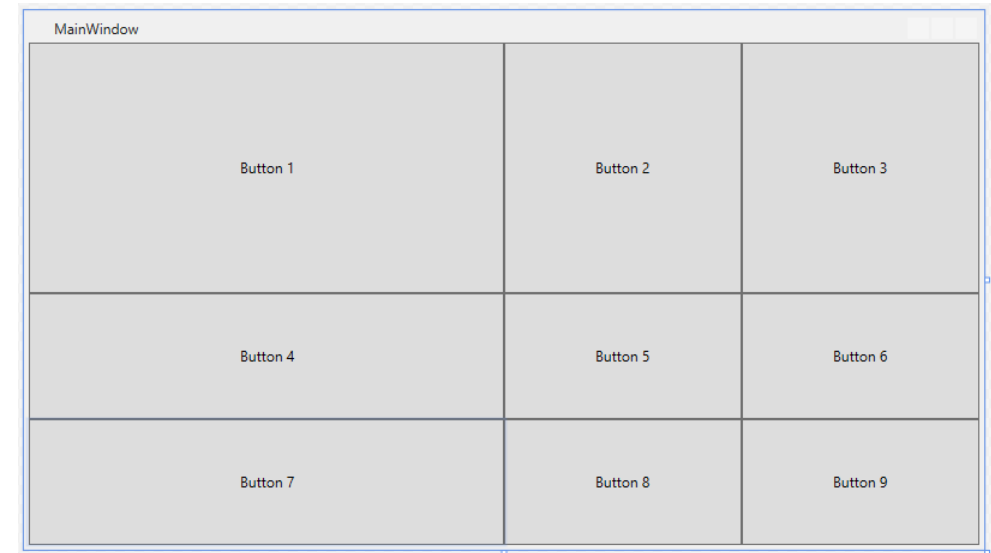


# Grid

```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="2*" />
    <ColumnDefinition Width="1*" />
    <ColumnDefinition Width="1*" />
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
    <RowDefinition Height="2*" />
    <RowDefinition Height="1*" />
    <RowDefinition Height="1*" />
  </Grid.RowDefinitions>
  <Button>Button 1</Button>
  <Button Grid.Column="1">Button 2</Button>
  <Button Grid.Column="2">Button 3</Button>
  <Button Grid.Row="1">Button 4</Button>
  <Button Grid.Column="1" Grid.Row="1">Button 5</Button>
  <Button Grid.Column="2" Grid.Row="1">Button 6</Button>
  <Button Grid.Row="2">Button 7</Button>
  <Button Grid.Column="1" Grid.Row="2">Button 8</Button>
  <Button Grid.Column="2" Grid.Row="2">Button 9</Button>
</Grid>
```

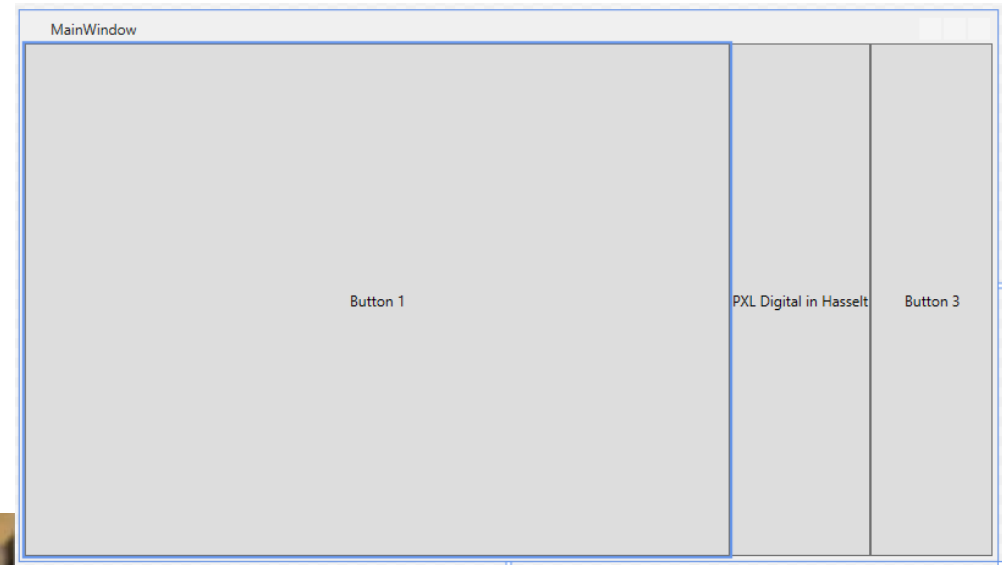
Neemt 2 maal  
zoveel plaats in  
qua kolommen

Neemt 2 maal  
zoveel plaats in  
qua rijen



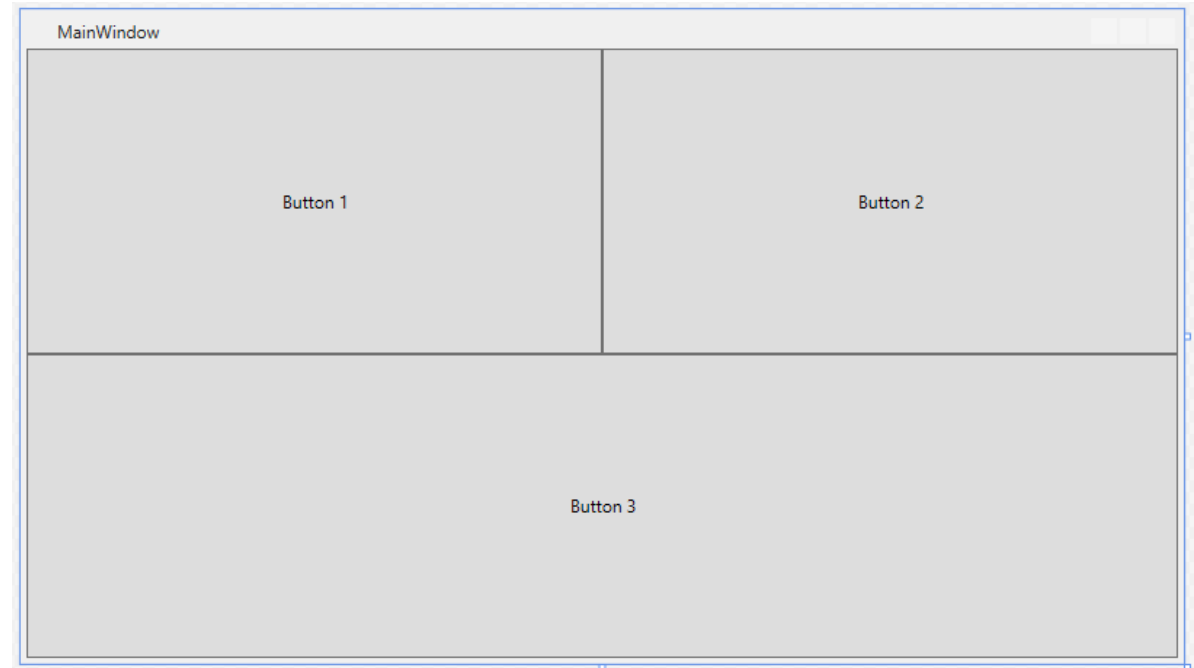
# Grid

```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="1*" />
    <ColumnDefinition Width="Auto" />
    <!--Breedte past zich aan de content van button aan-->
    <ColumnDefinition Width="100" />
  </Grid.ColumnDefinitions>
  <Button>Button 1</Button>
  <Button Grid.Column="1">PXL Digital in Hasselt</Button>
  <Button Grid.Column="2">Button 3</Button>
</Grid>
```



# Grid

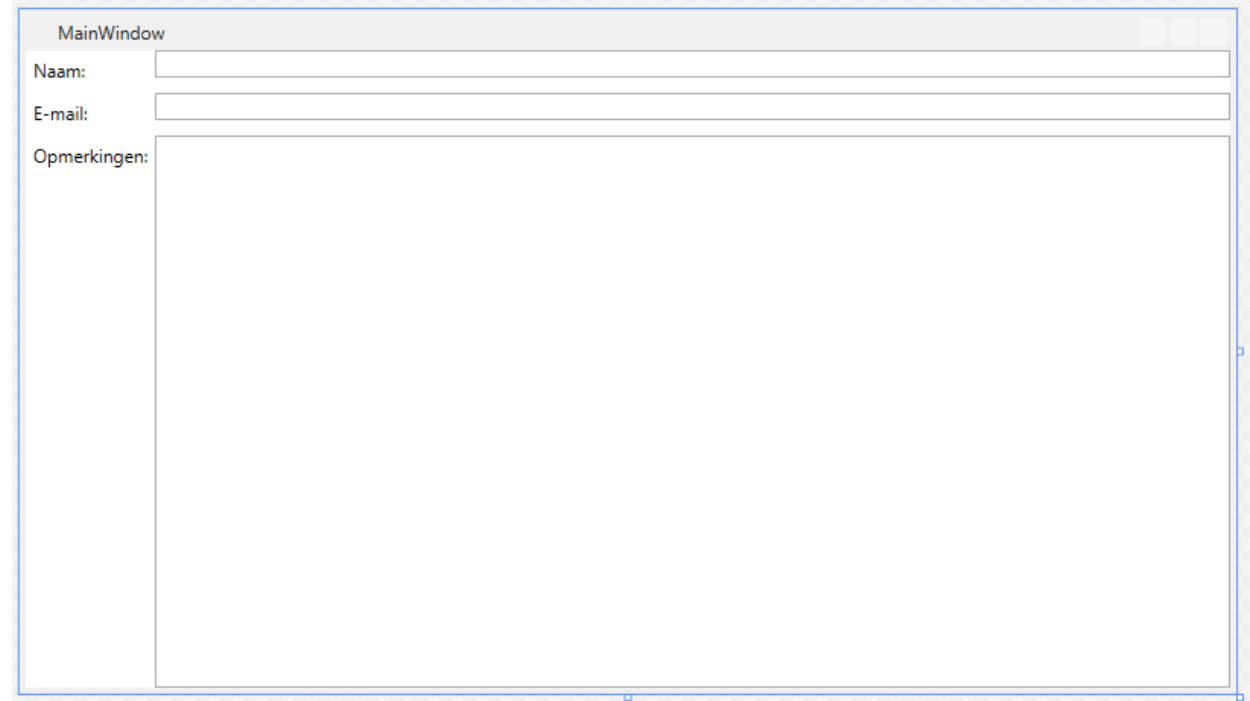
```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="1*" />
    <ColumnDefinition Width="1*" />
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
    <RowDefinition Height="*" />
    <RowDefinition Height="*" />
  </Grid.RowDefinitions>
  <Button>Button 1</Button>
  <Button Grid.Column="1">Button 2</Button>
  <Button Grid.Row="1" Grid.ColumnSpan="2">Button 3</Button>
  <!--Spanwijdte over 2 kolommen-->
</Grid>
```





# Grid

```
<Grid >
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="Auto" />
    <ColumnDefinition Width="*" />
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
    <RowDefinition Height="Auto" />
    <RowDefinition Height="Auto" />
    <RowDefinition Height="*" />
  </Grid.RowDefinitions>
  <Label>Naam:</Label>
  <TextBox Grid.Column="1" Margin="0,0,0,10" />
  <Label Grid.Row="1">E-mail:</Label>
  <TextBox Grid.Row="1" Grid.Column="1" Margin="0,0,0,10" />
  <Label Grid.Row="2">Opmerkingen:</Label>
  <TextBox Grid.Row="2" Grid.Column="1" AcceptsReturn="True" />
</Grid>
```



# Canvas

- Eenvoudigste paneel.
- Doet niets, groepeert gewoon componenten die er in staan.
  - Relatieve positie t.o.v. canvas via Canvas.Top, Canvas.Left,...
  - Als je canvas verplaatst, verplaats je ook elementen die in canvas zitten

```
<Canvas x:Name="CanTotaal" HorizontalAlignment="Left" Height="400" Margin="10,10,0,0" VerticalAlignment="Top"
        Width="400" Background="{DynamicResource {x:Static SystemColors.ControlDarkBrushKey}}">
    <Label Content="Totaal" HorizontalAlignment="Left" Height="35" VerticalAlignment="Top" Width="70"
          HorizontalContentAlignment="Right" Canvas.Left="10" Canvas.Top="10"></Label>
    <TextBox HorizontalAlignment="Left" Height="35" TextWrapping="Wrap" VerticalAlignment="Top" Width="70"
             Canvas.Left="10" Canvas.Top="50"></TextBox>
</Canvas>
```

# StackPanel

- Stapelt besturingselementen verticaal of horizontaal:
  - Orientation="Vertical" (standard)
    - Elementen krijgen dezelfde breedte (volgens breedste element).
  - Orientation="Horizontal"
    - Elementen krijgen dezelfde hoogte (volgens hoogste element).
- Als je wil dat besturingselementen volledige breedte of hoogte innemen.

# StackPanel

`<StackPanel>`

`<Button>Knop 1</Button>`

`<Button>Knop 2</Button>`

`<Button>Knop 3</Button>`

`<Button Width="140" Height="44">Knop 4</Button>`

`<Button>Knop 5</Button>`

`<Button>Knop 6</Button>`

`</StackPanel>`

`<StackPanel Orientation="Horizontal">`

`<Button>Knop 1</Button>`

`<Button>Knop 2</Button>`

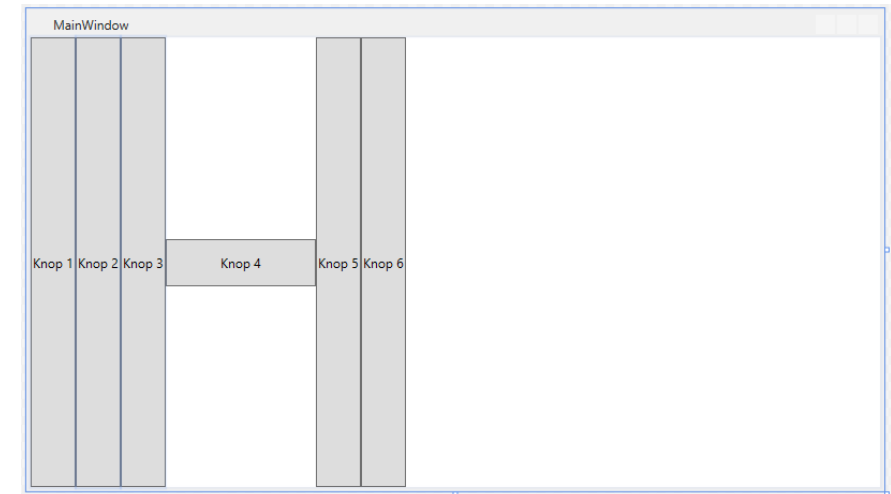
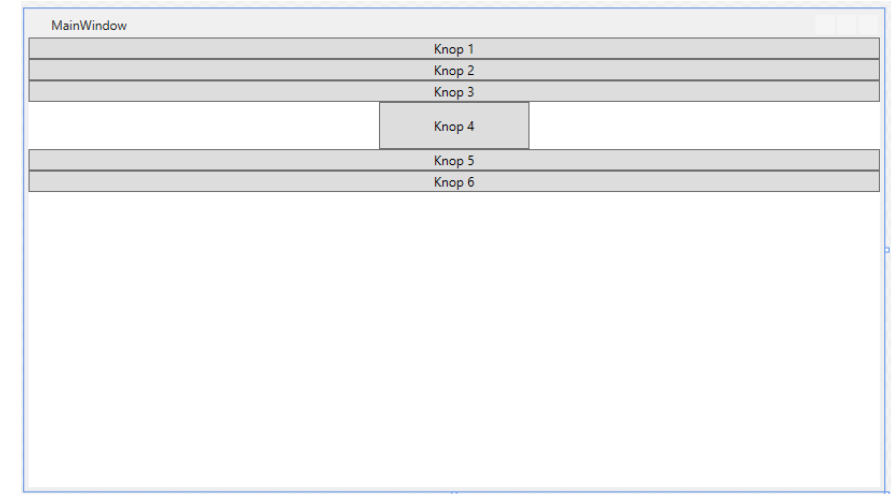
`<Button>Knop 3</Button>`

`<Button Width="140" Height="44">Knop 4</Button>`

`<Button>Knop 5</Button>`

`<Button>Knop 6</Button>`

`</StackPanel>`



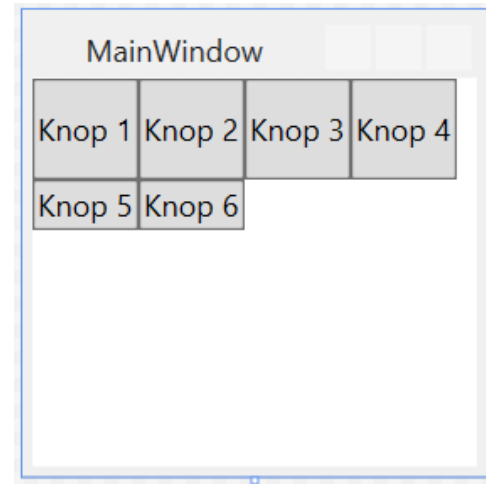


# WrapPanel

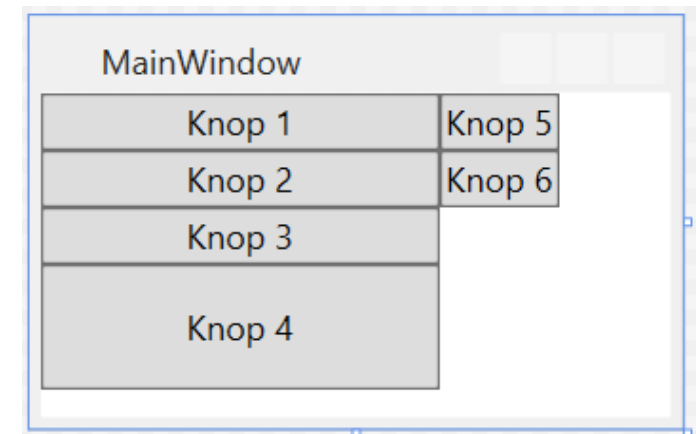
- Plaatst besturingselementen naast of onder elkaar
- Orientation="Horizontal" (standaard)
  - Wanneer rij vol  $\Rightarrow$  nieuwe rij starten
  - Controls krijgen dezelfde hoogte als hoogste element
- Orientation="Vertical"
  - Wanneer kolom vol  $\Rightarrow$  nieuwe kolom starten
  - Controls krijgen dezelfde breedte als breedste element

# WrapPanel

```
<WrapPanel>  
  <Button>Knop 1</Button>  
  <Button>Knop 2</Button>  
  <Button>Knop 3</Button>  
  <Button Height="40">Knop 4</Button>  
  <Button>Knop 5</Button>  
  <Button>Knop 6</Button>  
</WrapPanel>
```



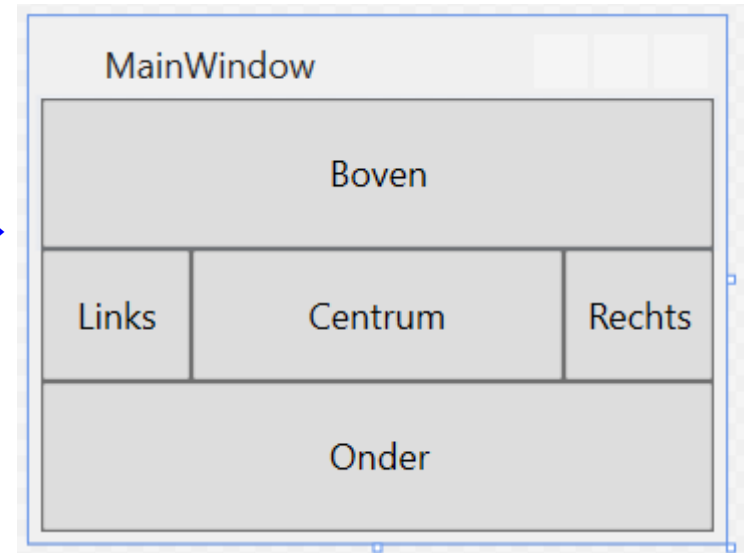
```
<WrapPanel Orientation="Vertical">  
  <Button>Knop 1</Button>  
  <Button>Knop 2</Button>  
  <Button>Knop 3</Button>  
  <Button Width="140" Height="44">Knop 4</Button>  
  <Button>Knop 5</Button>  
  <Button>Knop 6</Button>  
</WrapPanel>
```



# DockPanel

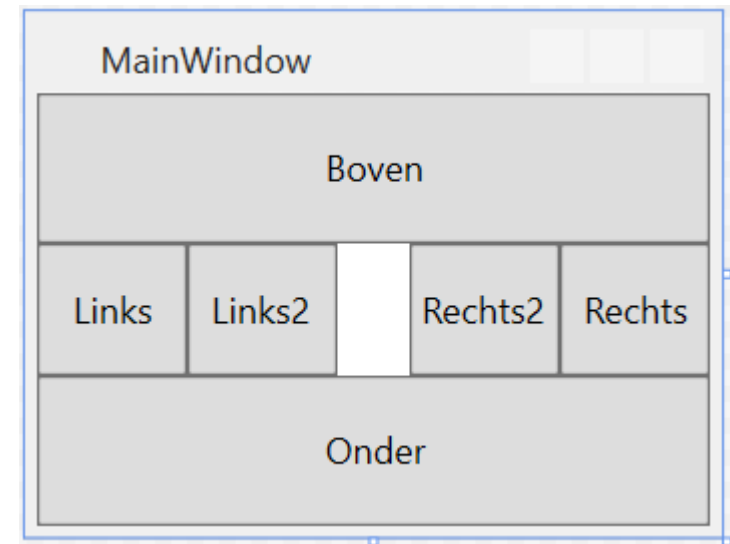
- Element boven, onder, links en rechts van elkaar plaatsen.
- Het eerste element komt standaard links, tenzij anders aangegeven.
- Het laatste element komt volledig gevuld in het midden.
  - Zet LastChildFill op false als je dit niet wil.

```
<DockPanel>  
  <Button DockPanel.Dock="Top" Height="50" Content="Boven"/>  
  <Button DockPanel.Dock="Bottom" Height="50" Content="Onder"/>  
  <Button DockPanel.Dock="Left" Width="50" Content="Links"/>  
  <Button DockPanel.Dock="Right" Width="50" Content="Rechts"/>  
  <Button Content="Centrum"/>  
</DockPanel>
```



# DockPanel

```
<DockPanel LastChildFill="false">  
  <Button DockPanel.Dock="Top" Height="50" Content="Boven"/>  
  <Button DockPanel.Dock="Bottom" Height="50" Content="Onder"/>  
  <Button DockPanel.Dock="Left" Width="50" Content="Links"/>  
  <Button DockPanel.Dock="Right" Width="50" Content="Rechts"/>  
  <Button DockPanel.Dock="Right" Width="50" Content="Rechts2"/>  
  <Button Width="50" Content="Links2"/>  
</DockPanel>
```



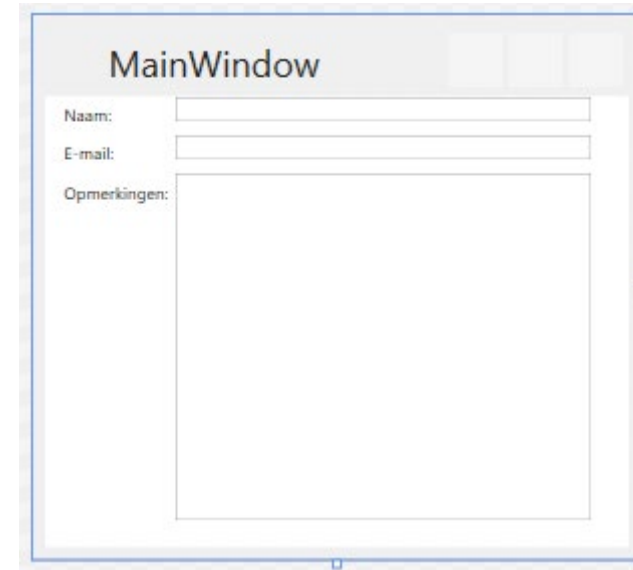


# ViewBox

- Past de inwendige elementen hun grootte aan naargelang wat er beschikbaar is.
- Formaat van inhoud aanpassen en schalen (transformatie).
- Vaak gebruikt voor 2D-afbeeldingen.

# ViewBox

```
<Viewbox>
  <Grid >
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="Auto" />
      <ColumnDefinition Width="*" />
      <ColumnDefinition Width="20" />
    </Grid.ColumnDefinitions>
    <Grid.RowDefinitions>
      <RowDefinition Height="Auto" />
      <RowDefinition Height="Auto" />
      <RowDefinition Height="*" />
      <RowDefinition Height="20" />
    </Grid.RowDefinitions>
    <Label>Naam:</Label>
    <TextBox Grid.Column="1" Margin="0,0,0,10" MinWidth="300"/>
    <Label Grid.Row="1">E-mail:</Label>
    <TextBox Grid.Row="1" Grid.Column="1" Margin="0,0,0,10" />
    <Label Grid.Row="2">Opmerkingen:</Label>
    <TextBox Grid.Row="2" Grid.Column="1" AcceptsReturn="True" MinHeight="250" />
  </Grid>
</Viewbox>
```



# ViewBox

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="98\*" />

<RowDefinition Height="97\*" />

<RowDefinition Height="138\*" />

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="316\*" />

<ColumnDefinition Width="271\*" />

</Grid.ColumnDefinitions>

<Button Content="Button 1" HorizontalAlignment="Left" Height="45"

Margin="69,39,0,0" VerticalAlignment="Top" Width="183" />

<Button Content="Button 2" HorizontalAlignment="Left" Height="42"

Margin="69,17,0,0" VerticalAlignment="Top" Width="181" Grid.Row="1" />

<Viewbox Margin="16,34,25,27" Grid.Column="1" Grid.RowSpan="2">

<Image Source="duimOmhoog.jpg" />

</Viewbox>

<Viewbox VerticalAlignment="Center" Grid.Row="2" Grid.RowSpan="2">

<TextBlock Text="Tekst in TextBlock is schaalbaar" />

</Viewbox>

</Grid>

