Windows Phone 7 - Tutoriel 18 : Horloge analogique

Horloge analogique utilisant Silverlight sur Windows Phone 7

Par Peter Bull - Deepin Prayag (Traduction) - Deepin (traducteur)

Date de publication : 19 avril 2012

Dernière mise à jour : 19 avril 2012

Ce tutoriel fait partie d'une **série** *d'articles de niveau débutant-intermédiaire pour apprendre Windows Phone 7 par le biais d'exemples pas à pas.*
Langage : C#
Public visé : niveau *Intermédiaire*
Commentez ce tutoriel : **Commentez** ♪

Traduction....3
Introduction....3
Horloge analogique utilisant Silverlight sur Windows Phone 7....3
Étape 1....3
Étape 2....3
Étape 3....4
Étape 4....5
Étape 5....5
Étape 6....6
Étape 7....7
Étape 8....9
Étape 9....12
Étape 10....13
Étape 11....16
Étape 12....17
Étape 13....17
Étape 14....18
Étape 15....19
Conclusion....19
Liens....19
Remerciements....20

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Traduction

Ce tutoriel est la traduction la plus fidèle possible du tutoriel original de **Peter Bull**, **Windows Phone 7 - Analogue Clock using Silverlight on Windows Phone 7**.

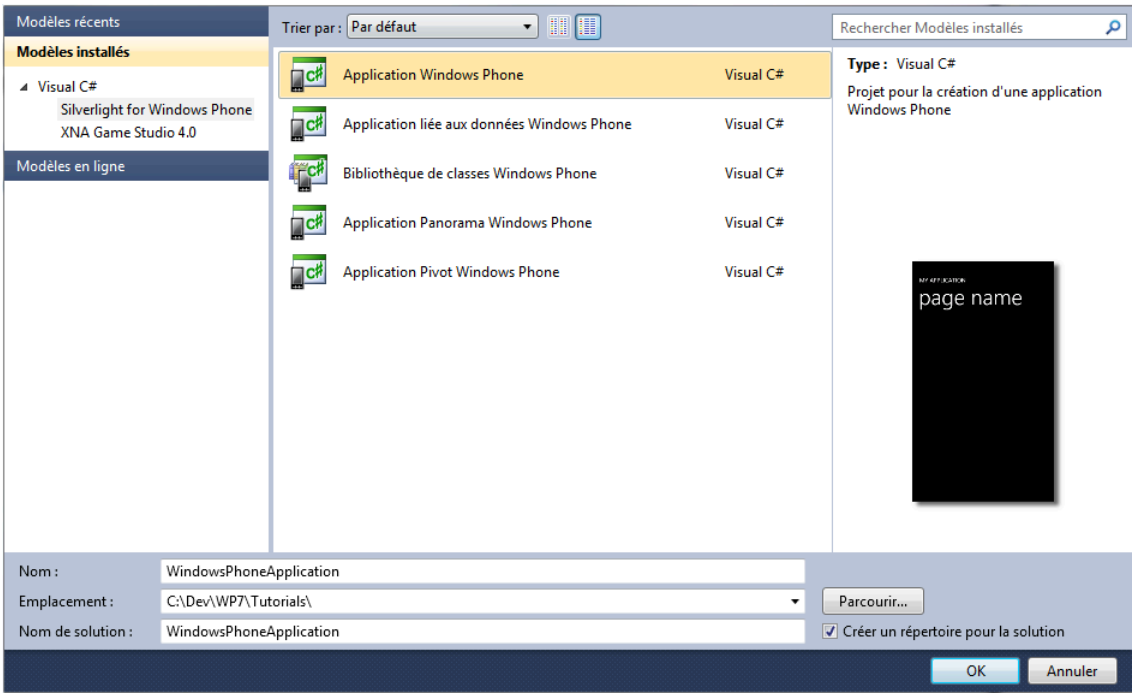
Introduction

Analogue Clock est un contrôle utilisateur simple pour afficher l'heure actuelle en utilisant Silverlight sur Windows Phone 7.

Horloge analogique utilisant Silverlight sur Windows Phone 7

Étape 1

Démarrez Microsoft Visual Web Developer 2010 Express pour Windows Phone, ensuite sélectionnez *Fichier* puis *Nouveau Projet…* Sélectionnez *Visual C#* ensuite *Silverlight for Windows Phone* puis *Application Windows Phone* dans les modèles installés, sélectionnez un emplacement si vous le souhaitez, puis entrez un nom pour le projet et appuyez sur OK :



Étape 2

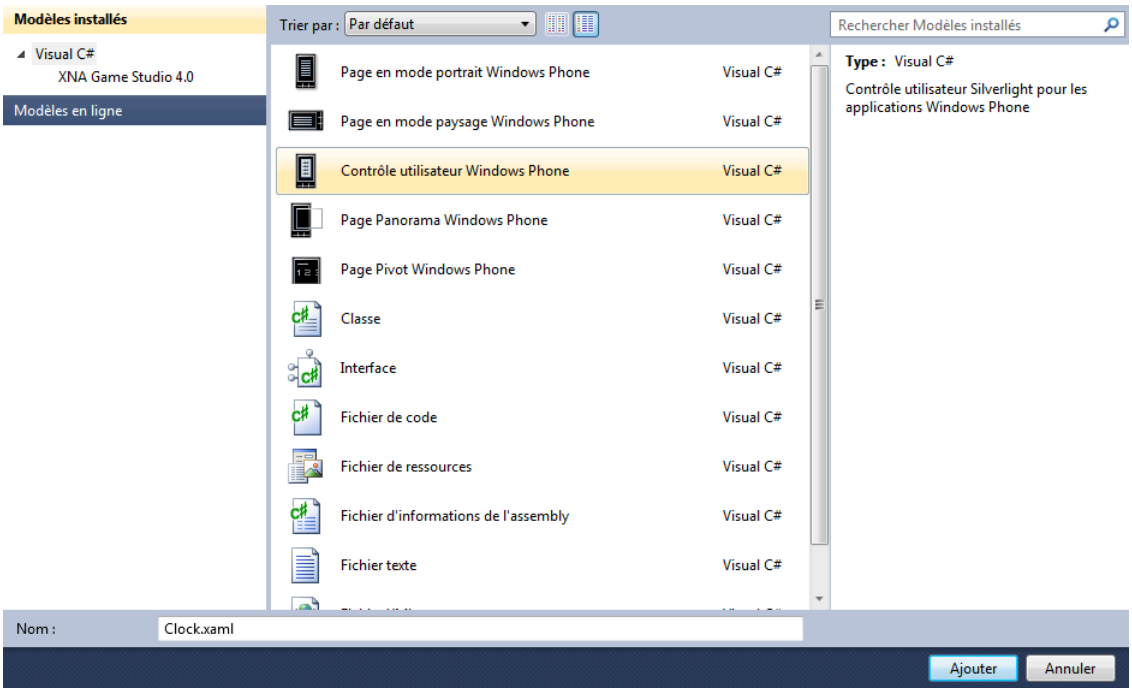
Une page *Application Windows Phone* nommée *MainPage.xaml* apparaît :

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Étape 3

Sélectionnez *Projet* puis *Ajouter un nouvel élément…* et sélectionnez le modèle *Contrôle utilisateur Windows Phone*, ensuite modifiez le *Nom* en *Clock.xaml :*

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/



Étape 4

Ajoutez le nouveau *contrôle utilisateur* au *Projet* en cliquant sur *Ajouter*, ensuite dans le volet XAML pour le *contrôle utilisateur* enlevez la ligne `Background="{StaticResource PhoneChromeBrush}"` :

```
<UserControl x:Class="AnalogueClock.Clock"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    FontFamily="{StaticResource PhoneFontFamilyNormal}"
    FontSize="{StaticResource PhoneFontSizeNormal}"
    Foreground="{StaticResource PhoneForegroundBrush}"
    d:DesignHeight="480" d:DesignWidth="480">

    <Grid x:Name="LayoutRoot">

    </Grid>
</UserControl>
```

Étape 5

Toujours dans le volet XAML, entre les lignes `<Grid x:Name="LayoutRoot">` et `</Grid>` tapez le code XAML suivant :

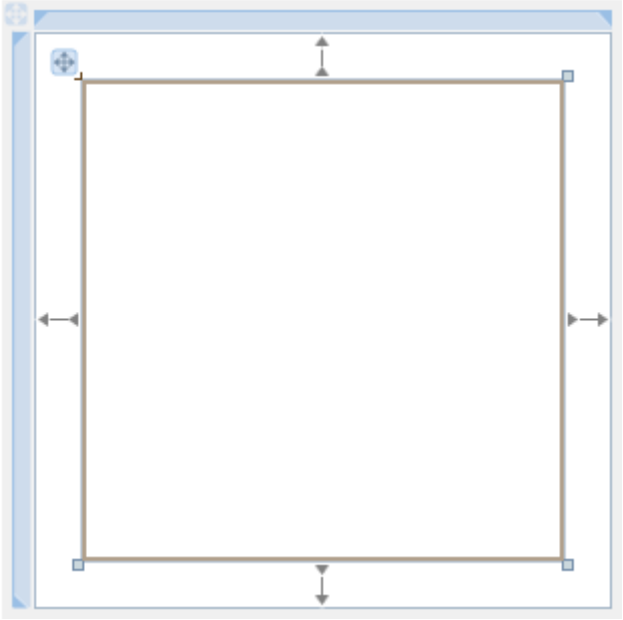
```
<Canvas Height="400" Width="400" Name="Display"/>
```

XAML :

```
<Grid x:Name="LayoutRoot">
    <Canvas Height="400" Width="400" Name="Display"/>
</Grid>
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Design :



Étape 6

Faites un clic droit sur le *contrôle utilisateur* ou sur l'entrée pour *Clock.xaml* dans *l'Explorateur de solutions* et choisissez l'option *Afficher le code*. Dans la *vue de code* pour *Clock.xaml.cs*, au-dessus de namespace AnalogueClock tapez le code suivant :

```
using System.Windows.Threading;
```

Toujours dans la *vue de code*, en dessous du « { » de la ligne public partial class Clock : UserControl tapez ceci :

```
private DispatcherTimer _timer = new DispatcherTimer();
private DateTime _time = DateTime.Now;
private Canvas _markers = new Canvas();
private Canvas _face = new Canvas();
private Rectangle _secondsHand;
private Rectangle _minutesHand;
private Rectangle _hoursHand;
private double _diameter;
private int _secondsWidth = 1;
private int _secondsHeight;
private int _minutesWidth = 5;
private int _minutesHeight;
private int _hoursWidth = 8;
private int _hoursHeight;
private bool _isRealTime = true;
private bool _showSeconds = true;
private bool _showMinutes = true;
private bool _showHours = true;
private Brush _faceBackground = ((Brush)App.Current.Resources["PhoneBackgroundBrush"]);
private Brush _faceForeground = ((Brush)App.Current.Resources["PhoneForegroundBrush"]);
private Brush _rimForeground = ((Brush)App.Current.Resources["PhoneBorderBrush"]);
private Brush _rimBackground = ((Brush)App.Current.Resources["PhoneAccentBrush"]);
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Animation;
using System.Windows.Shapes;
using System.Windows.Threading;

namespace AnalogueClock
{
    public partial class Clock : UserControl
    {
        private DispatcherTimer _timer = new DispatcherTimer();
        private DateTime _time = DateTime.Now;
        private Canvas _markers = new Canvas();
        private Canvas _face = new Canvas();
        private Rectangle _secondsHand;
        private Rectangle _minutesHand;
        private Rectangle _hoursHand;
        private double _diameter;
        private int _secondsWidth = 1;
        private int _secondsHeight;
        private int _minutesWidth = 5;
        private int _minutesHeight;
        private int _hoursWidth = 8;
        private int _hoursHeight;
        private bool _isRealTime = true;
        private bool _showSeconds = true;
        private bool _showMinutes = true;
        private bool _showHours = true;
        private Brush _faceBackground = ((Brush)App.Current.Resources["PhoneBackgroundBrush"]);
        private Brush _faceForeground = ((Brush)App.Current.Resources["PhoneForegroundBrush"]);
        private Brush _rimForeground = ((Brush)App.Current.Resources["PhoneBorderBrush"]);
        private Brush _rimBackground = ((Brush)App.Current.Resources["PhoneAccentBrush"]);

        public Clock()
        {
            InitializeComponent();
        }
    }
}
```

Étape 7

Pendant que nous en sommes à la *vue de code* pour *Clock.xaml.cs*, au-dessus de la ligne `public Clock()`, tapez les *méthodes* suivantes :

```
private Rectangle Hand(double width, double height,
Brush background, double radiusX, double radiusY, double thickness)
{
  Rectangle _hand = new Rectangle();
  _hand.Width = width;
  _hand.Height = height;
  _hand.Fill = background;
  _hand.StrokeThickness = thickness;
  _hand.RadiusX = radiusX;
  _hand.RadiusY = radiusY;
  return _hand;
}

private void RemoveHand(ref Rectangle hand)
{
  if (hand != null && _face.Children.Contains(hand))
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
    {
      _face.Children.Remove(hand);
    }
  }

  private void AddHand(ref Rectangle hand)
  {
    if (!_face.Children.Contains(hand))
    {
      _face.Children.Add(hand);
    }
  }

  private TransformGroup TransformGroup(double Angle, double X, double Y)
  {
    TransformGroup _transformGroup = new TransformGroup();
    TranslateTransform _firstTranslate = new TranslateTransform();
    _firstTranslate.X = X;
    _firstTranslate.Y = Y;
    _transformGroup.Children.Add(_firstTranslate);
    RotateTransform _rotateTransform = new RotateTransform();
    _rotateTransform.Angle = Angle;
    _transformGroup.Children.Add(_rotateTransform);
    TranslateTransform _secondTranslate = new TranslateTransform();
    _secondTranslate.X = _diameter / 2;
    _secondTranslate.Y = _diameter / 2;
    _transformGroup.Children.Add(_secondTranslate);
    return _transformGroup;
  }
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
private Rectangle Hand(double width, double height,
Brush background, double radiusX, double radiusY, double thickness)
{
    Rectangle _hand = new Rectangle();
    _hand.Width = width;
    _hand.Height = height;
    _hand.Fill = background;
    _hand.StrokeThickness = thickness;
    _hand.RadiusX = radiusX;
    _hand.RadiusY = radiusY;
    return _hand;
}

private void RemoveHand(ref Rectangle hand)
{
    if (hand != null && _face.Children.Contains(hand))
    {
        _face.Children.Remove(hand);
    }
}

private void AddHand(ref Rectangle hand)
{
    if (!_face.Children.Contains(hand))
    {
        _face.Children.Add(hand);
    }
}

private TransformGroup TransformGroup(double Angle, double X, double Y)
{
    TransformGroup _transformGroup = new TransformGroup();
    TranslateTransform _firstTranslate = new TranslateTransform();
    _firstTranslate.X = X;
    _firstTranslate.Y = Y;
    _transformGroup.Children.Add(_firstTranslate);
    RotateTransform _rotateTransform = new RotateTransform();
    _rotateTransform.Angle = Angle;
    _transformGroup.Children.Add(_rotateTransform);
    TranslateTransform _secondTranslate = new TranslateTransform();
    _secondTranslate.X = _diameter / 2;
    _secondTranslate.Y = _diameter / 2;
    _transformGroup.Children.Add(_secondTranslate);
    return _transformGroup;
}
```

Étape 8

Toujours dans la *vue de code* pour *Clock.xaml.cs*, au-dessus de la ligne `public Clock()`, tapez les *méthodes* suivantes :

```
private void SecondHand(int seconds)
{
  RemoveHand(ref _secondsHand);
  if (_showSeconds)
  {
    _secondsHand = Hand(_secondsWidth,_secondsHeight,_faceForeground,0, 0, 0);
    _secondsHand.RenderTransform = TransformGroup(seconds * 6,
    -_secondsWidth / 2, -_secondsHeight + 4.25);
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
      AddHand(ref _secondsHand);
    }
  }

  private void MinuteHand(int minutes, int seconds)
  {
    RemoveHand(ref _minutesHand);
    if (_showMinutes)
    {
      _minutesHand = Hand(_minutesWidth, _minutesHeight, _rimForeground, 2, 2, 0.6);
      _minutesHand.RenderTransform = TransformGroup(6 * minutes + seconds / 10,
      -_minutesWidth / 2, -_minutesHeight + 4.25);
      AddHand(ref _minutesHand);
    }
  }

  private void HourHand(int hours, int minutes, int seconds)
  {
    RemoveHand(ref _hoursHand);
    if (_showHours)
    {
      _hoursHand = Hand(_hoursWidth, _hoursHeight, _rimBackground, 3, 3, 0.6);
      _hoursHand.RenderTransform = TransformGroup(30 * hours + minutes / 2 + seconds / 120,
      -_hoursWidth / 2, -_hoursHeight + 4.25);
      AddHand(ref _hoursHand);
    }
  }

  private void Layout(ref Canvas canvas)
  {
    Ellipse _rim = new Ellipse();
    Ellipse _inner = new Ellipse();
    canvas.Children.Clear();
    _diameter = canvas.Width;
    _rim.Height = _diameter;
    _rim.Width = _diameter;
    _rim.Fill = _rimBackground;
    canvas.Children.Add(_rim);
    _inner.Width = _diameter - 40;
    _inner.Height = _diameter - 40;
    _inner.Fill = _faceBackground;
    Canvas.SetTop(_inner, 20);
    Canvas.SetLeft(_inner, 20);
    canvas.Children.Add(_inner);
    _markers.Children.Clear();
    _markers.Width = _diameter;
    _markers.Height = _diameter;
    for (int i = 0; i < 60; i++)
    {
      Rectangle _marker = new Rectangle();
      _marker.Fill = _rimForeground;
      if ((i % 5) == 0)
      {
        _marker.Width = 3;
        _marker.Height = 8;
        _marker.RenderTransform = TransformGroup(i * 6, -(_marker.Width / 2),
        -(_marker.Height * 2 + 4.5 - _rim.StrokeThickness / 2 - _diameter / 2 - 6));
      }
      else
      {
        _marker.Width = 1;
        _marker.Height = 4;
        _marker.RenderTransform = TransformGroup(i * 6, -(_marker.Width / 2),
        -(_marker.Height * 2 + 12.75 - _rim.StrokeThickness / 2 - _diameter / 2 - 8));
      }
      _markers.Children.Add(_marker);
    }
    canvas.Children.Add(_markers);
    _face.Width = _diameter;
    _face.Height = _diameter;
    canvas.Children.Add(_face);
    _secondsHeight = (int)_diameter / 2 - 20;
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
    _minutesHeight = (int)_diameter / 2 - 40;
    _hoursHeight = (int)_diameter / 2 - 60;
}
```

```
        private void SecondHand(int seconds)
        {
            RemoveHand(ref _secondsHand);
            if (_showSeconds)
            {
                _secondsHand = Hand(_secondsWidth, _secondsHeight, _faceForeground, 0, 0, 0);
                _secondsHand.RenderTransform = TransformGroup(seconds * 6,
                -_secondsWidth / 2, -_secondsHeight + 4.25);
                AddHand(ref _secondsHand);
            }
        }

        private void MinuteHand(int minutes, int seconds)
        {
            RemoveHand(ref _minutesHand);
            if (_showMinutes)
            {
                _minutesHand = Hand(_minutesWidth, _minutesHeight, _rimForeground, 2, 2, 0.6);
                _minutesHand.RenderTransform = TransformGroup(6 * minutes + seconds / 10,
                -_minutesWidth / 2, -_minutesHeight + 4.25);
                AddHand(ref _minutesHand);
            }
        }

        private void HourHand(int hours, int minutes, int seconds)
        {
            RemoveHand(ref _hoursHand);
            if (_showHours)
            {
                _hoursHand = Hand(_hoursWidth, _hoursHeight, _rimBackground, 3, 3, 0.6);
                _hoursHand.RenderTransform = TransformGroup(30 * hours + minutes / 2 + seconds / 120,
                -_hoursWidth / 2, -_hoursHeight + 4.25);
                AddHand(ref _hoursHand);
            }
        }
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
        private void Layout(ref Canvas canvas)
        {
            Ellipse _rim = new Ellipse();
            Ellipse _inner = new Ellipse();
            canvas.Children.Clear();
            _diameter = canvas.Width;
            _rim.Height = _diameter;
            _rim.Width = _diameter;
            _rim.Fill = _rimBackground;
            canvas.Children.Add(_rim);
            _inner.Width = _diameter - 40;
            _inner.Height = _diameter - 40;
            _inner.Fill = _faceBackground;
            Canvas.SetTop(_inner, 20);
            Canvas.SetLeft(_inner, 20);
            canvas.Children.Add(_inner);
            _markers.Children.Clear();
            _markers.Width = _diameter;
            _markers.Height = _diameter;
            for (int i = 0; i < 60; i++)
            {
                Rectangle _marker = new Rectangle();
                _marker.Fill = _rimForeground;
                if ((i % 5) == 0)
                {
                    _marker.Width = 3;
                    _marker.Height = 8;
                    _marker.RenderTransform = TransformGroup(i * 6, -(_marker.Width / 2),
                    -(_marker.Height * 2 + 4.5 - _rim.StrokeThickness / 2 - _diameter / 2 - 6));
                }
                else
                {
                    _marker.Width = 1;
                    _marker.Height = 4;
                    _marker.RenderTransform = TransformGroup(i * 6, -(_marker.Width / 2),
                    -(_marker.Height * 2 + 12.75 - _rim.StrokeThickness / 2 - _diameter / 2 - 8));
                }
                _markers.Children.Add(_marker);
            }
            canvas.Children.Add(_markers);
            _face.Width = _diameter;
            _face.Height = _diameter;
            canvas.Children.Add(_face);
            _secondsHeight = (int)_diameter / 2 - 20;
            _minutesHeight = (int)_diameter / 2 - 40;
            _hoursHeight = (int)_diameter / 2 - 60;
        }
```

Étape 9

Toujours dans la *vue de code* pour *Clock.xaml.cs*, dans le *constructeur* `public Clock()`, en dessous de `InitializeComponent();` tapez ceci :

```
Layout(ref Display);
_timer.Tick += (object sender, EventArgs e) =>
{
  if (_isRealTime) _time = DateTime.Now;
  SecondHand(_time.Second);
  MinuteHand(_time.Minute, _time.Second);
  HourHand(_time.Hour, _time.Minute, _time.Second);
};
_timer.Start();
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
public Clock()
{
    InitializeComponent();
    Layout(ref Display);
    _timer.Tick += (object sender, EventArgs e) =>
    {
        if (_isRealTime) _time = DateTime.Now;
        SecondHand(_time.Second);
        MinuteHand(_time.Minute, _time.Second);
        HourHand(_time.Hour, _time.Minute, _time.Second);
    };
    _timer.Start();
}
```

Étape 10

Toujours dans la *vue de code* pour *Clock.xaml.cs*, au-dessus de la ligne public Clock(), tapez les *propriétés* suivantes :

```
public Brush FaceForeground
{
  get { return _faceForeground; }
  set
  {
    _faceForeground = value == null ?
    ((Brush)App.Current.Resources["PhoneForegroundBrush"]) :
    value;
  }
}

public Brush FaceBackground
{
  get { return _faceBackground; }
  set
  {
    _faceBackground = value == null ?
    ((Brush)App.Current.Resources["PhoneBackgroundBrush"]) :
    value;
  }
}

public Brush RimForeground
{
  get { return _rimForeground; }
  set
  {
    _rimForeground = value == null ?
    ((Brush)App.Current.Resources["PhoneAccentBrush"]) :
    value;
  }
}

public Brush RimBackground
{
  get { return _rimBackground; }
  set
  {
    _rimBackground = value == null ?
    ((Brush)App.Current.Resources["PhoneSubtleBrush"]) :
    value;
  }
}

public bool Enabled
{
  get { return _timer.IsEnabled;}
  set
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
    {
      if (_timer.IsEnabled)
      {
        _timer.Stop();
      }
      else
      {
        _timer.Start();
      }
    }
  }

  public bool IsRealTime
  {
    get {return _isRealTime; }
    set {_isRealTime = value;}
  }

  public bool ShowSeconds
  {
    get {return _showSeconds; }
    set {_showSeconds = value;}
  }

  public bool ShowMinutes
  {
    get {return _showMinutes; }
    set {_showMinutes = value;}
  }

  public bool ShowHours
  {
    get {return _showHours; }
    set {_showHours = value;}
  }

  public DateTime Time
  {
    get {return _time;}
    set {_time = value;}
  }
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

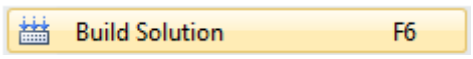
```
        public Brush FaceForeground
        {
            get { return _faceForeground; }
            set
            {
                _faceForeground = value == null ?
                ((Brush)App.Current.Resources["PhoneForegroundBrush"]) :
                value;
            }
        }

        public Brush FaceBackground
        {
            get { return _faceBackground; }
            set
            {
                _faceBackground = value == null ?
                ((Brush)App.Current.Resources["PhoneBackgroundBrush"]) :
                value;
            }
        }

        public Brush RimForeground
        {
            get { return _rimForeground; }
            set
            {
                _rimForeground = value == null ?
                ((Brush)App.Current.Resources["PhoneAccentBrush"]) :
                value;
            }
        }

        public Brush RimBackground
        {
            get { return _rimBackground; }
            set
            {
                _rimBackground = value == null ?
                ((Brush)App.Current.Resources["PhoneSubtleBrush"]) :
                value;
            }
        }
```


Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

```
        public bool Enabled
        {
            get { return _timer.IsEnabled; }
            set
            {
                if (_timer.IsEnabled)
                {
                    _timer.Stop();
                }
                else
                {
                    _timer.Start();
                }
            }
        }

        public bool IsRealTime
        {
            get { return _isRealTime; }
            set { _isRealTime = value; }
        }

        public bool ShowSeconds
        {
            get { return _showSeconds; }
            set { _showSeconds = value; }
        }

        public bool ShowMinutes
        {
            get { return _showMinutes; }
            set { _showMinutes = value; }
        }

        public bool ShowHours
        {
            get { return _showHours; }
            set { _showHours = value; }
        }

        public DateTime Time
        {
            get { return _time; }
            set { _time = value; }
        }
```

Étape 11

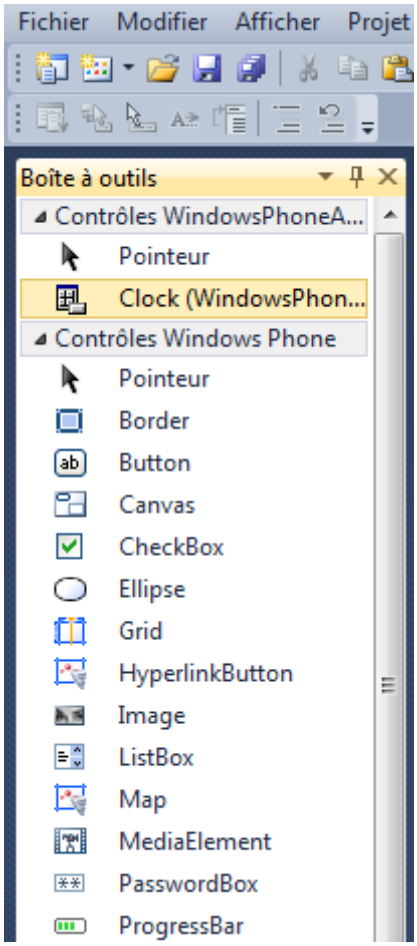
Sélectionnez *Build Solution* depuis le menu *Debug :*



Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Étape 12

Quand le *Build* est terminé, retournez au *concepteur de vues MainPage* en sélectionnant l'onglet *MainPage.xaml,* puis dans la section *Contrôles WindowsPhoneAnalogueClock* de la *Boite à outils*, sélectionnez le contrôle *Clock :*



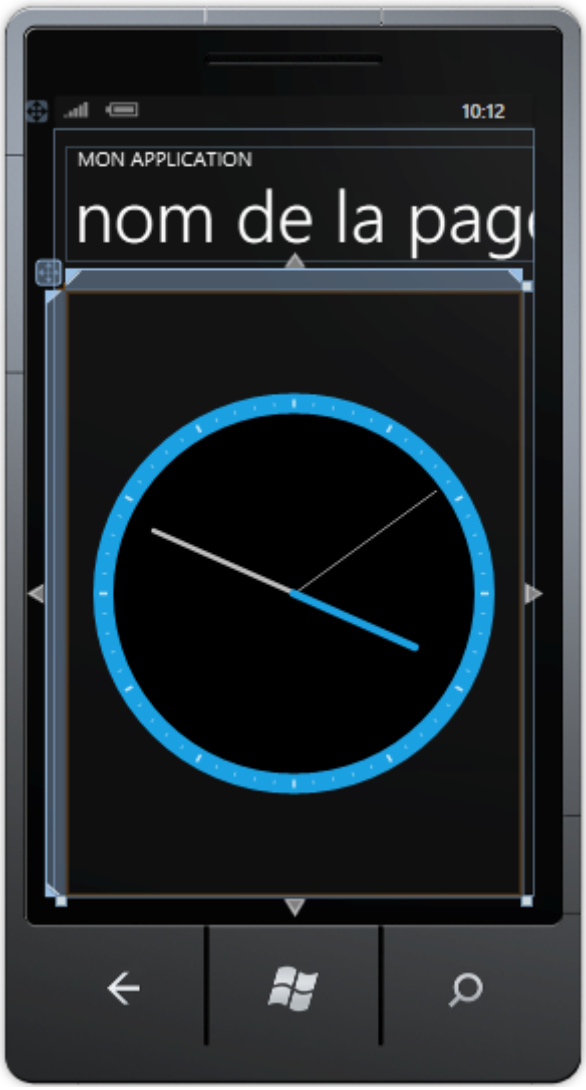
Étape 13

Ajoutez un objet *Clock* sur la page et dans le volet XAML pour *MainPage.xaml* et modifiez la ligne *Clock1* comme ceci :

```
<my:Clock x:Name="Clock"/>
```

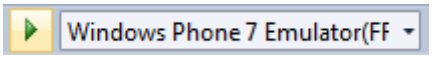
Voir ci-dessous :

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/



Étape 14

Enregistrez le *Projet* maintenant que vous avez terminé l'application *Windows Phone Silverlight*. Sélectionnez l'option *Windows Phone Emulator* ensuite sélectionnez *Debug* puis *Démarrer le débogage* ou cliquez sur *Démarrer le débogage :*



Une fois l'application chargée, la vue suivante apparaît dans l'*émulateur Windows Phone :*

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Étape 15

Vous pouvez ensuite *Arrêter l'application* en sélectionnant la fenêtre d'application *Visual Studio 2010* et en cliquant sur le bouton *Arrêter le débogage :*

Conclusion

Il s'agit d'une horloge analogique simple. Vous pouvez modifier les couleurs utilisées en sélectionnant l'horloge. Allez ensuite dans la boite de Propriétés, trouvez les propriétés *FaceForeground*, *RimBackground,* etc.et modifiez-les. Vous pouvez également personnaliser l'horloge afin qu'elle utilise des dégradés de couleur ou d'autres effets - personnalisez-la !

Liens

Télécharger le code source du tutoriel.

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/

Remerciements

Je tiens ici à remercier **Peter Bull** de m'avoir autorisé à traduire son tutoriel.
Je remercie **tomlev** pour sa relecture technique et ses propositions.
Je remercie également **ram-0000** pour sa relecture orthographique et ses propositions.

Copyright ® 2012 Peter Bull. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://deepin.developpez.com/tutoriels/windows-phone/silverlight/horloge-analogique/