

Tiles for UWP

2018.03.07 更新

グレースィティ株式会社

目次

Tiles for UWP	2
主な特長	3
クイックスタート	4
手順1: タイルアプリケーションの設定	4-5
手順2: アプリケーションへのタイルの追加	5-6
手順3: タイルアプリケーションの実行	6-7
C1Tiles の使い方	8
C1Tile コントロール	8-9
C1FlipTile コントロール	9
C1SlideTile コントロール	9-10
ContentTemplate と BackContentTemplate	10-11
タスク別ヘルプ	12
タイルヘッダーに XAML コンテンツを追加する	12
タイルコンテンツを更新する	12-13
項目のコレクションに連結する	13
連結コントロールでのタイルの使用	13-14

Tiles for UWP

Tiles for UWP を使用して、さまざまなタイル表示やナビゲーションハブを作成できます。タイルを使用すると、Windows 8 のスタート画面の操作性を独自のアプリケーションに容易に採り入れることができます。Live 更新によるスライドおよび反転アニメーションをサポートするさまざまなタイルコントロールを利用できます。タイルをさまざまなコンテナと組み合わせることで、多種多様なレイアウトを作成することが可能です。

主な特長

Tiles for UWP には、次の主な特長があります。

- **反転およびスライドタイトルの作成**

C1FlipTile コントロールと **C1SlideTile** コントロールを使用して、コンテンツをスライドアニメーションで交互に表示するタイトルを作成できます。テンプレートを設計し、コントロールにコンテンツを提供するだけです。更新とアニメーションは自動的に処理されます。

- **おなじみの Windows 8 ライブタイトルの動作**

Tiles は、Windows ストアアプリおよび Windows Phone アプリ用に特別に設計されています。どのタイトルコントロールも、Windows 8 や Windows Phone のスタート画面にあるライブタイトルと同じ形式のインタラクティブな動作を示します。したがって、タイトルを選択するとき以外は指を上げる必要がないという、ユーザーにとってはおなじみの動作でアプリを操作できます。

- **Live 更新**

Tiles では、最新の「ライブ」コンテンツをフリップ、スライド、および表示できます。静的 **C1TileService** クラスと **UpdateInterval** プロパティを使用して、更新間隔を簡単に制御できます。詳細については、「[タイトルコンテンツを更新する](#)」を参照してください。

- **任意のコンテナでホスト可能**

Tiles は、**C1TileListBox**、**C1WrapPanel**、標準の **GridView** および **ListBox** コントロールなど、任意の **ItemsControl** コンテナでホストできます。したがって、**C1Tile** をデータ連結シナリオで使用できます。コンテナごとに異なる方法で複数の **C1Tile** を配置できるため、それらの組み合わせと可能性は無限です。

- **異なるサイズのサポート**

すべてのタイトルを同じ大きさで作成する必要はありません。さまざまな種類やサイズのタイトルを組み合わせ、それぞれのアプリケーションで個性的な表示方法を作成できます。

クイックスタート

このクイックスタートガイドは、**Tiles for UWP** を初めて使用するユーザーのために用意されています。このクイックスタートでは、Visual Studio で新しいプロジェクトを作成し、アプリケーションに **Tiles for UWP** コントロールを追加して、コントロールの外観と動作をカスタマイズします。

C1Tile、C1SlideTile、および C1FlipTile コントロールを含むアプリケーションを作成します。各コントロールの詳細については、「**C1Tile コントロール**」、「**C1SlideTile コントロール**」、および「**C1FlipTile コントロール**」を参照してください。

手順1: タイルアプリケーションの設定

この手順では、最初に Visual Studio で **Tiles for UWP** を使用するアプリケーションを作成します。この手順では、Windows ストアアプリケーションを作成し、C1TileListBox を追加し、**Tiles for UWP** コントロールを挿入します。それには、次の手順に従います。

1. Visual Studio で、**[ファイル]**→**[新規作成]**→**[プロジェクト]**を選択します。
2. **[新しいプロジェクト]**ダイアログボックスで、左ペインの言語を展開し、言語の下で**[Windows ストア]**を選択し、テンプレートリストで**[新しいアプリケーション (XAML)]**を選択します。プロジェクトの名前を入力し、**[OK]**をクリックしてプロジェクトを作成します。
MainPage.xaml ページが表示されます。
3. <Page> タグの直後<Grid> タグの前に、次のマークアップを追加します。

▶ XAML でマークアップの書き方

マークアップ

```
<Page.Resources>
    <Style x:Key="listBoxItemStyle" TargetType="Xaml:C1ListBoxItem">
        <Setter Property="Tile:C1TileService.PointerDownAnimation"
            Value="True"/>
    </Style>
    <Style TargetType="Tile:C1Tile" x:Key="baseTileStyle">
        <Setter Property="Background" Value="#FFC410" />
        <Setter Property="Foreground" Value="White"/>
        <Setter Property="FontSize" Value="80"/>
        <Setter Property="HeaderForeground" Value="White"/>
        <Setter Property="HeaderFontSize" Value="12"/>
        <Setter Property="BorderThickness" Value="0" />
        <Setter Property="Width" Value="280" />
        <Setter Property="Height" Value="200" />
    </Style>
    <Style TargetType="Tile:C1Tile" BasedOn="{StaticResource
baseTileStyle}">
    </Style>
    <Style TargetType="Tile:C1SlideTile" BasedOn="{StaticResource
baseTileStyle}">
    </Style>
    <Style TargetType="Tile:C1FlipTile" BasedOn="{StaticResource
baseTileStyle}">
    </Style>
</Page.Resources>
```

このマークアップは、アプリケーションの外観をスタイル設定するためのページリソースを追加します。

4. <Grid> タグと </Grid> タグの間にカーソルを置き、ツールボックスに移動し、**C1TileListBox** コントロールをダブルクリックしてページに追加します。これにより、C1.Xaml アセンブリへの参照も追加されます。
5. **C1TileListBox** マークアップを更新します。次のようになります。

マークアップ

```
<Xaml:C1TileListBox ItemStyle="{StaticResource listBoxItemStyle}">
    <Xaml:C1TileListBox.Items>
    </Xaml:C1TileListBox.Items>
</Xaml:C1TileListBox>
```

この手順では、Windows ストアアプリケーションを作成しました。次の手順では、このアプリケーションに **Tiles for UWP** コントロールを追加します。

手順2: アプリケーションへのタイルの追加

前の手順では、Windows ストアアプリケーションを作成しました。この手順では、**Tiles for UWP** コントロールを追加します。次の手順に従います。

1. <cl:C1TileListBox.Items></cl:C1TileListBox.Items> タグの間にカーソルを置き、ツールボックスに移動し、C1Tile コントロールをダブルクリックしてページに追加します。これにより、**C1.Xaml.Tile** アセンブリへの参照も追加されます。
2. **C1Tile** マークアップを更新します。次のようになります。

マークアップ

```
<Tile:C1Tile Content="1" Header="C1Tile" HeaderPadding="12" Padding="0"
    HeaderBackground="#22000000" HorizontalHeaderAlignment="Stretch" />
```

このマークアップは、ヘッダーとコンテンツテキストの追加、パディングの追加、ヘッダーの背景色と配置の変更を行います。

3. **C1Tile** マークアップの直後に次のマークアップを追加します。

マークアップ

```
<Tile:C1SlideTile Content="2" HeaderPadding="12" HorizontalContentAlignment="Stretch"
    VerticalContentAlignment="Stretch" Padding="0"
    Header="C1SlideTile">
    <Tile:C1Tile.ContentTemplate>
    <DataTemplate>
        <Border Background="#FFBC1C48" >
            <TextBlock Text="{Binding}" VerticalAlignment="Center"
                HorizontalAlignment="Center" />
        </Border>
    </DataTemplate>
</Tile:C1Tile.ContentTemplate>
<Tile:C1Tile.BackContentTemplate>
    <DataTemplate>
        <Border Background="#FF028541" >
            <StackPanel VerticalAlignment="Center" HorizontalAlignment="Center">
                <TextBlock Text="{Binding}" Foreground="White" HorizontalAlignment="Center"/>
                <TextBlock Text="Back Content Template" Margin="0 -10 0 0" FontSize="12"
                    Foreground="White" HorizontalAlignment="Center"/>
            </StackPanel>
        </Border>
    </DataTemplate>
</Tile:C1Tile.BackContentTemplate>
</Tile:C1SlideTile>
```

このマークアップは、**ContentTemplate** および **BackContentTemplate** と共に **C1SlideTile** コントロールを追加します。**ContentTemplate** は、コントロールの初期コンテンツと外観を決定します。**BackContentTemplate** は、コントロールが変化するとき(C1SlideTile コントロールの場合は、1つのテンプレートから別のテンプレートにスライドするとき)、コントロールのコンテンツと外観を決定します。

4. **C1SlideTile** マークアップの直後に次のマークアップを追加します。

マークアップ

```
<Tile:C1SlideTile Content="3" BackContent="Back Content 3" FontSize="36" Header="C1SlideTile"
    HeaderPadding="12" Padding="0"/>
    <Tile:C1SlideTile Content="4" Header="C1SlideTile, SlideDirection = Right"
        HeaderPadding="12" Padding="0"
        SlideDirection="Right" HorizontalHeaderAlignment="Right"/>
    <Tile:C1SlideTile Content="5" Header="C1SlideTile" />
    <Tile:C1FlipTile Content="6" Header="C1FlipTile" />
    <Tile:C1FlipTile Content="7" Header="C1FlipTile" HeaderPadding="12" Padding="0"
        HeaderBackground="#22000000">
        <Tile:C1Tile.ContentTemplate>
    <DataTemplate>
```

```

<Border Background="#FF8C0095" >
    <TextBlock Text="{Binding}" VerticalAlignment="Center"
        HorizontalAlignment="Center" />
</Border>
</DataTemplate>
</Tile:C1Tile.ContentTemplate>
<Tile:C1Tile.BackContentTemplate>
    <DataTemplate>
        <Border Background="#FFCD4900" >
            <StackPanel VerticalAlignment="Center" HorizontalAlignment="Center">
                <TextBlock Text="{Binding}" Foreground="White" HorizontalAlignment="Center"/>
                <TextBlock Text="Back Content Template" Margin="0 -10 0 0" FontSize="12"
                    Foreground="White" HorizontalAlignment="Center"/>
            </StackPanel>
        </Border>
    </DataTemplate>
</Tile:C1Tile.BackContentTemplate>
</Tile:C1FlipTile>

```

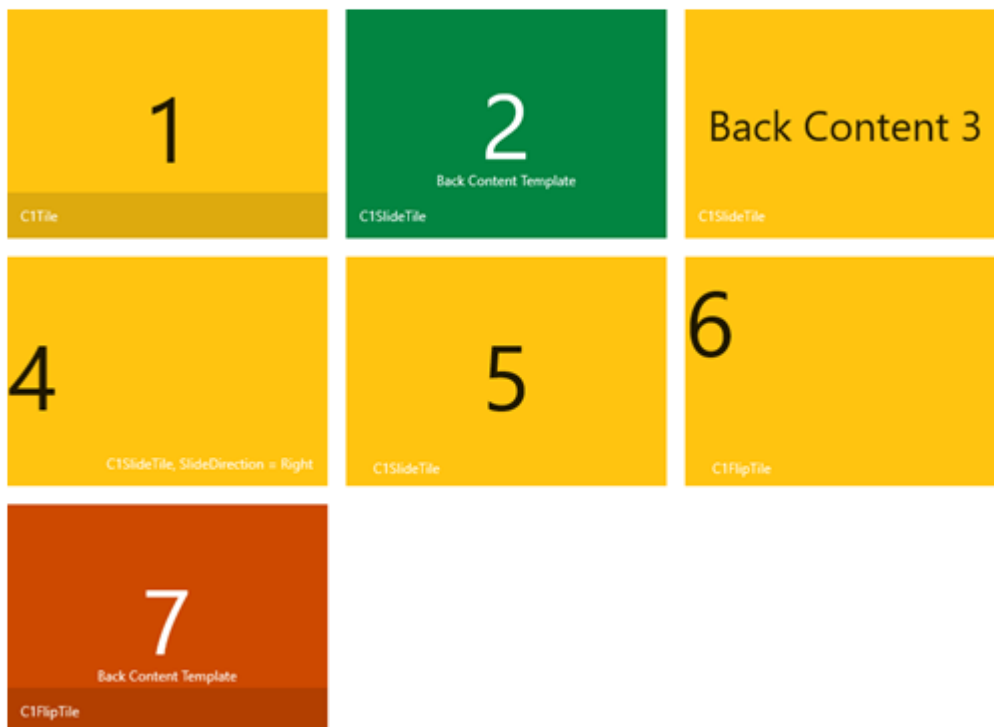
5. このマークアップは、**C1SlideTile** コントロールと **C1FlipTile** コントロールを追加します。**C1FlipTile** コントロールの 1 つには、**ContentTemplate** と **BackContentTemplate** も含まれます。

この手順では、アプリケーションに **Tiles for UWP** コントロールを追加しました。次の手順では、アプリケーションを実行し、実行時にアプリケーションがどのように表示されるかを確認します。

手順3: タイルアプリケーションの実行

前の手順では、Windows ストアアプリケーションを作成し、**Tiles for UWP** コントロールを追加してカスタマイズしました。この手順ではアプリケーションを実行します。

1. **[デバッグ]** → **[デバッグ開始]** を選択して、アプリケーションを実行します。次のようになります。



上の図の緑色の **C1SlideTile** は、**ContentTemplate** を使用して表示されます。オレンジ色の **C1FlipTile** は、**BackContentTemplate** を使用して表示されます。

2. 各コントロールの動作を確認してください。**C1SlideTile** はコンテンツの選択項目間をスライドするように表示され、**C1FlipTile** はフリップするように表示されます。
3. いくつかの **C1SlideTile** コントロールは、異なる方向に移動するように表示されます。この動作は、**SlideDirection** プロパティを使用して設定できます。

Tiles for UWP

おめでとうございます。これで **Tiles for UWP** クイックスタートは終了です。**Tiles for UWP** コントロールの使用方法の詳細については、以降のトピックを参照してください。

C1Tiles の使い方

以下のトピックでは、**Tiles for UWP** の操作の概要を説明します。

- C1Tileコントロール
- C1FlipTileコントロール
- C1SlideTileコントロール
- ContentTemplates と BackContentTemplate

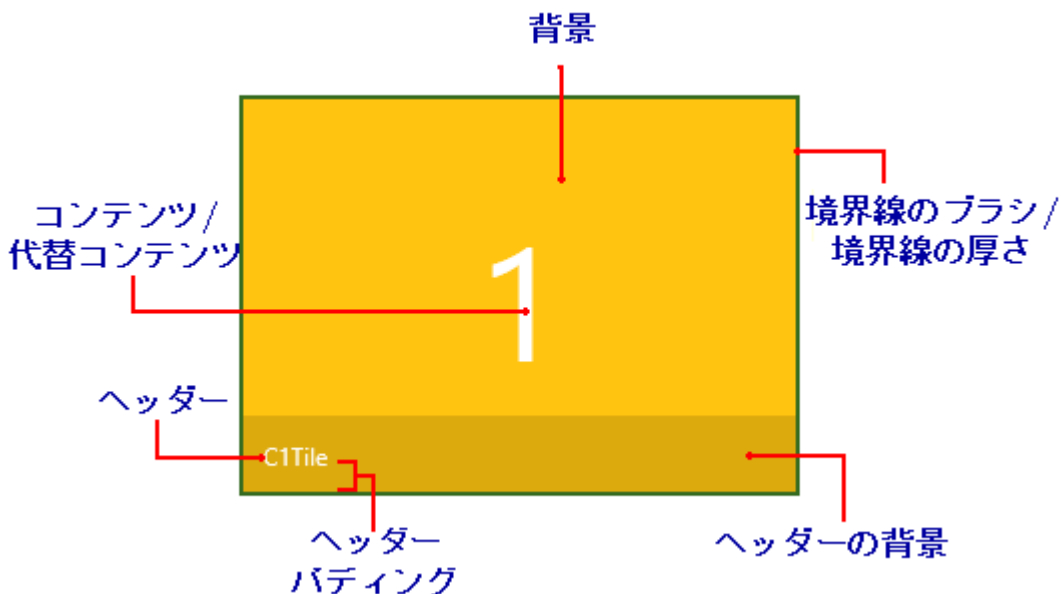
C1Tile コントロール

C1Tile コントロールは、Windows 8 のライブタイルの動作を真似たヘッダー付きのアニメーションコンテンツコントロールです。たとえば、以下のマークアップは、いくつかのプロパティセットを含む基本的な C1Tile コントロールを作成します。

マークアップ

```
<Tile:C1Tile Content="1" BackContent="1" Header="C1Tile" HeaderBackground="#22000000" HeaderPadding="12"
Padding="0" BorderBrush="#FF356A21" BorderThickness="2" Background="#FFC410" Foreground="White"
FontSize="80" HeaderForeground="White" HeaderFontSize="12" Width="280" Height="200"/>
```

次の図に、上のマークアップの結果と設定されたプロパティの一部を示します。



上の図とマークアップでは、次のプロパティが設定されています。

- **Content**: このプロパティは、C1Tile コントロールの初期コンテンツを設定します。上の図では、**Content** が「1」に設定されています。
- **BackContent**: このプロパティは、C1Tile コントロールの代替コンテンツを設定します。上の図では、**BackContent** も「1」に設定されています (Content と同じ)。ただし、**BackContent** プロパティに別の値が設定された場合は、タイルのコンテンツが変化したときにその値が表示されます。
- **Header**: このプロパティは、C1Tile コントロールのヘッダーのコンテンツを設定します。デフォルトでは、ヘッダーはタイルの下部に表示されます。上の図では、**Header** は「C1Tile」に設定されています。
- **HeaderBackground**: このプロパティは、ヘッダーの背景の色と透過性を設定します。上の例では、**HeaderBackground** プロパティは 22000000 に設定されています。最初の2つの値は色の透過性を示し、後の6つの値は色が黒であることを示しています。
- **HeaderPadding**: **HeaderPadding** プロパティは、**Header** 値の周囲のパディングを設定します。上の例では、**HeaderPadding** が「12」に設定されています。数値が大きいほど、**Header** プロパティで表される値がタイルの

Tiles for UWP

端から離れて表示されます。

- **Padding: Padding** プロパティは、タイル内の **Content** と **Header** の周囲のパディングを設定します。上の例では、**Padding** が「0」に設定されています。これは、ヘッダー領域がタイルの下境界線にぴったりくっついて表示されることを示します。数値が大きいほど、**Content** プロパティと **Header** プロパティで表される値がタイルの端から離れて表示されます。
- **BorderBrush: BorderBrush** プロパティは、C1Tile コントロールの周囲の境界線の色を示します。上の例では、これは「#FF356A21」(不透明な緑色)に設定されています。
- **BorderThickness: BorderThickness** は、C1Tile を囲む境界線の厚さを示します。上の例では、このプロパティは「2」に設定されています。

C1FlipTile コントロール

C1FlipTile コントロールは、C1Tile コントロールに基づいています。ただし、C1FlipTile コントロールは、コンテンツが変化する際のアニメーション効果が異なります。C1FlipTile コントロールでは、アニメーションによってタイルがコンテンツ間をフリップしているように見えます。

C1FlipTile コントロールのマークアップの例を以下に示します。

マークアップ

```
<Tile:C1FlipTile Content="7" Header="C1FlipTile" HeaderBackground="#22000000" HeaderPadding="12"
Padding="0" BorderThickness="2" Background="#8B008B" Foreground="White" FontSize="80"
HeaderForeground="White" HeaderFontSize="12" VerticalContentAlignment="Center"
HorizontalContentAlignment="Center" Width="280" Height="200"/>
```

上のマークアップは、次の C1FlipTile コントロールを作成します。



C1SlideTile コントロール

C1Tile コントロールに基づく **C1SlideTile** コントロールには、テンプレート間をスライドするアニメーションが含まれます。このコンテンツのスライド方向は、**SlideDirection** プロパティによって決まります。**SlideDirection** プロパティは、次のいずれかの値に設定できます(デフォルト値は **All**)。

値	方向
Up	下から上。
Left	右から左。
Down	上から下。
Right	左から右。
All(デフォルト)	すべて。

たとえば、**SlideDirection** プロパティを **Right** に設定した **C1SlideTile** コントロールの例を以下に示します。

マークアップ

```
<Tile:C1SlideTile Content="4" Header="C1SlideTile, SlideDirection = Right"
Padding="0"
    SlideDirection="Right" HorizontalHeaderAlignment="Right" Background="#FFC410"
Foreground="White" FontSize="80" HeaderForeground="White" HeaderFontSize="12"
Width="280" Height="200"/>
```

上のマークアップは、次の **C1SlideTile** コントロールを作成します。



ContentTemplate と BackContentTemplate

ContentTemplate と **BackContentTemplate** を使用すると、**Tiles for UWP** コントロールの外観をカスタマイズすることができます。タイルに最初に表示するコンテンツを **ContentTemplate** に置き、別のコンテンツにスライドまたはフリップされたときに表示するコンテンツを **BackContentTemplate** に置きます。

たとえば、次のマークアップは、**ContentTemplate** と **BackContentTemplate** を **C1SlideTile** コントロールに追加します。

実行すると、最初は **ContentTemplate** が次のように表示されます。

マークアップ

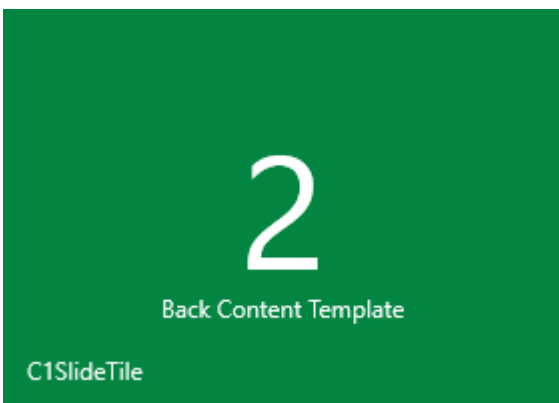
```
<Tile:C1SlideTile Content="2" HeaderPadding="12" HorizontalContentAlignment="Stretch"
VerticalContentAlignment="Stretch" Padding="0"
Header="C1SlideTile" Foreground="White" FontSize="80" HeaderForeground="White"
HeaderFontSize="12" Width="280" Height="200">
    <Tile:C1Tile.ContentTemplate>
        <DataTemplate>
            <Border Background="#FFBC1C48" >
                <TextBlock Text="{Binding}" VerticalAlignment="Center"
HorizontalAlignment="Center" />
            </Border>
        </DataTemplate>
    </Tile:C1Tile.ContentTemplate>
    <Tile:C1Tile.BackContentTemplate>
        <DataTemplate>
            <Border Background="#FF028541" >
                <StackPanel VerticalAlignment="Center"
HorizontalAlignment="Center">
                    <TextBlock Text="{Binding}" Foreground="White"
```

Tiles for UWP

```
HorizontalAlignment="Center"/>
        <TextBlock Text="Back Content Template" Margin="0 -10 0
0" FontSize="12" Foreground="White"
        HorizontalAlignment="Center"/>
    </StackPanel>
</Border>
</DataTemplate>
</Tile:C1Tile.BackContentTemplate>
</Tile:C1SlideTile>
```



もう一方のコンテンツテンプレートに変化すると、コントロールには次の **BackContentTemplate** が表示されます。



タスク別ヘルプ

タスク別ヘルプは、ユーザーの皆様が Visual Studio のプログラミングに精通しており、Windows ストアアプリケーションを作成して使用する一般的な方法を理解していることを前提としています。**Tiles for UWP** 製品に精通していない場合は、まず「**クイックスタート**」を参照してください。

各トピックは、**Tiles for UWP** 製品を使用して特定のタスクを実行するためのソリューションを提供します。ヘルプに示される手順に従って作業を進めるだけで、さまざまな **Tiles for UWP** の機能を具体的に示すプロジェクトを作成できます。

次のトピックは、新しい Windows ストアプロジェクトが作成されていることを前提としています。

タイトルヘッダーに XAML コンテンツを追加する

Tiles for UWP コントロールは、**Header** プロパティに直接設定した XAML コンテンツをサポートしません。**Header** を何らかの CLR 値に連結するか、**Header** は空にしておき、Content テンプレートに目的の **Header** コンテンツ(画像など)を追加する必要があります。

次に例を示します。

マークアップ

```
<ctile:C1FlipTile Width="280" Height="200" Header="{Binding Name}" />
```

タイトルコンテンツを更新する

C1SlideTile コントロールと **C1FlipTile** コントロールは、コンテンツを自動的に一定の間隔で交互に表示します。この間隔は、静的 **C1TileService** クラスと **UpdateInterval** プロパティを使用して調整できます。次に例を示します。

▶ Visual Basic コードの書き方

Visual Basic

```
C1TileService.UpdateInterval = TimeSpan.FromSeconds(20)
```

▶ C# コードの書き方

C#

```
C1TileService.UpdateInterval = TimeSpan.FromSeconds(20);
```

更新間隔の値を大きくすると、更新の頻度が低くなります。

プログラムで、各タイトルの **UpdateTile** メソッドを呼び出して、タイトルを更新することもできます。たとえば、クリックされたときにタイトルを更新するには、次のコードを使用します。

▶ Visual Basic コードの書き方

Visual Basic

```
Private Sub C1Tile_Click(sender As Object, e As System.EventArgs)
    Dim tile As C1Tile = TryCast(sender, C1Tile)
    If tile IsNot Nothing Then
        tile.UpdateTile()
    End If
End Sub
```

▶ C# コードの書き方

```
C#
private void C1Tile_Click(object sender, System.EventArgs e)
{
    C1Tile tile = sender as C1Tile;
    if(tile != null)
        tile.UpdateTile();
}
```

項目のコレクションに連結する

C1SlideTile または **C1FlipTile** コントロールでは、2つ以上の項目を交互に表示することができます。**ContentSource** プロパティを使用して、フリップする項目をいくつでも指定することができます。**ContentTemplate** (およびオプションで **AlternateContentTemplate**) を定義して、連結コンテンツの外観を定義します。

マークアップ

```
<c1:C1SlideTile Header="Photos" ContentSource="{Binding Items}">
  <c1:C1SlideTile.ContentTemplate>
    <DataTemplate>
      <Grid>
        <TextBlock Text="{Binding Author}" Foreground="White"
VerticalAlignment="Top" Margin="4,2,0,2"/>
        <Image Source="{Binding Thumbnail}" Stretch="UniformToFill"
Margin="24, 24, 1, 1"/>
      </Grid>
    </DataTemplate>
  </c1:C1SlideTile.ContentTemplate>
</c1:C1SlideTile>
```

連結コントロールでのタイルの使用

C1TileListBox や **GridView** コントロールなどの任意の **ItemsControl** で **C1Tile** を使用できます。連結 **ListBox** コントロールで **C1FlipTile** を使用するマークアップの例を以下に示します。


マークアップ

```
<ListBox x:Name="listBox" ItemsSource="{Binding}">
  <ListBox.ItemsPanel>
    <ItemsPanelTemplate>
      <Xaml:C1WrapPanel
Background="YellowGreen"/>
    </ItemsPanelTemplate>
  </ListBox.ItemsPanel>
  <ListBox.ItemContainerStyle>
    <Style TargetType="ListBoxItem">
      <Setter
Property="Tile:C1TileService.PointerDownAnimation" Value="True"/>
    </Style>
  </ListBox.ItemContainerStyle>
```

```

        <ListBox.ItemTemplate>
            <DataTemplate>
                <Tile:C1FlipTile Height="200"
                Width="200" Header="{Binding Title}" Content="{Binding}"
                Background="DarkGreen" HeaderBackground="#88000000"
                HeaderFontSize="18" Command="{Binding
                TileCommand, ElementName=pageRoot}" HeaderPadding="2" Padding="0"
                HeaderForeground="White"
                CommandParameter="{Binding Content,
                RelativeSource={RelativeSource Self}}">
                    <Tile:C1Tile.ContentTemplate>
                        <DataTemplate>
                            <Grid>
                                <TextBlock Text="{Binding Author}" Foreground="White"
                                VerticalAlignment="Top" Margin="4,2,0,2"/>
                                <Image Source="{Binding Thumbnail}"
                                Stretch="UniformToFill" Margin="1, 24, 24, 1"/>
                            </Grid>
                        </DataTemplate>
                    </Tile:C1Tile.ContentTemplate>
                    <Tile:C1Tile.BackContentTemplate>
                        <DataTemplate>
                            <Grid>
                                <Image Source="{Binding Thumbnail}"
                                Stretch="UniformToFill"/>
                            </Grid>
                        </DataTemplate>
                    </Tile:C1Tile.BackContentTemplate>
                </Tile:C1FlipTile>
            </DataTemplate>
        </ListBox.ItemTemplate>
    </ListBox>

```

 **注意:** `VirtualizingStackPanel` を `ItemsPanel` として使用し、`ListBox` で `C1Tile` コントロールを使用する場合は、`VirtualizingStackPanel.VirtualizationMode` を `Standard` に設定して、`ListBox` のスクロール中にタイルアニメーションが発生しないようにしてください。