

洗練された Windows フォーム

Win32 APIのテクニックはもういらない。
コードを大幅に削減できるWindows フォームの魅力

Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:

Level



Samples

はじめに

VB6.0では、フォームの機能が弱く、少し高度なことをしたければ、Win32 APIの手を借りてカスタマイズする実装が必要でした。

しかしVB.NETでは、フォームが高機能になり、Win32 APIを使うことなく、ほぼすべての処理を.NET Framework クラスライブラリだけで実装できます。

VB6.0とVB.NETでは、フォームが備えるメソッドやプロパティの名前が変わっただけでなく、操作が変わったものや廃止されたコントロールなどもあり、いくつかのロジックは、作り直しが必要です。

そこで本稿では、VB6.0との違いも含めながら、VB.NETにおけるフォームの扱い方を全般的に説明します。

dows.Forms.Formクラスから継承するクラスとして構成されます。

VB6.0と違い、「.frm」のような専用の拡張子はなく、フォームの拡張子も、他のクラスと同様に、「.vb」の拡張子を持ちます。

Windows フォームは [プロジェクト] メニューの [Windows フォームの追加] をクリックすると、いくつでも作成できます (図1)。

図1を見るとわかるように、「ダイアログボックス」という選択肢はありませんが、ダイアログボックスも Windows フォームとして扱われるため、ダイアログボックスを作る場合でも、「Windows フォーム」を選んで作成し、あとで境界線を設定するFormBorderStyleなどのプロパティを変更して、ダイアログボックスの形状に変更してゆくことになります。

フォームの基本

VB.NETにおけるフォームは、「Windows フォーム」と呼ばれ、System.Win

フォームの設計

フォームは、「フォームデザイナ」を使って、ツールボックスから必要なコントロールをドラッグ&ドロップして

図1：Windows フォームの追加



貼り付けていきます。また、プロパティウィンドウで、必要なプロパティを変更していきます。

このような開発方法は、VB6.0と同じですが、フォームにコントロールを貼り付けたり、フォームやコントロー

ルのプロパティを変更したりすると、それに伴い、コードが自動生成されるという点が大きく違います。

VB.NETでは、フォームやコントロールを初期化するコードが#Region～#End Regionで囲まれた部分に記述さ

れます。

#Region～#End Regionで書かれた部分は、Visual Studio .NETでは通常隠されており、

[+] Windows フォーム デザイナで生成されたコード

と表示されています。この [+] 部分をクリックして開くと、リスト1のようなプログラムが現われます。

フォームを初期化するコードは、InitializeComponent メソッド内にあります。

```
<System.Diagnostics.DebuggerStepThrough> _
Private Sub InitializeComponent()
    ' Me.Name = "Form1" などプロパティを
    ' 初期化するコード
End Sub
```

この部分は、フォームデザイナーでフォームやコントロールを変更すると自動的に書き換わります。そのため、開

リスト1：Visual Studio .NETによって自動生成されるコード

```
#Region " Windows フォーム デザイナで生成されたコード "
```

```
Public Sub New()
    MyBase.New()
```

```
' この呼び出しは Windows フォーム デザイナが必要です。
InitializeComponent()
```

```
' InitializeComponent() 呼び出しの後に初期化を追加します。
' ——①ここに必要な初期化コードを追記する——
```

```
End Sub
```

```
' Form は、コンポーネント一覧に後処理を実行
' するために dispose をオーバーライドします。
Protected Overloads Overrides Sub Dispose( _
    ByVal disposing As Boolean)
    ' ——②ここに必要な終了コードを追記する——
```

```
If disposing Then
    If Not (components Is Nothing) Then
        components.Dispose()
    End If
End If
```

```
MyBase.Dispose(disposing)
```

```
End Sub
```

```
' Windows フォーム デザイナが必要です。
Private components As System.ComponentModel.IContainer
```

```
' メモ：以下のプロシージャは、Windows フォーム デザイナ
' が必要です。
```

```
' Windows フォーム デザイナを使って変更してください。
' コード エディタを使って変更しないでください。
```

```
<System.Diagnostics.DebuggerStepThrough> _
Private Sub InitializeComponent()
```

```
'
```

```
'Form1
```

```
'
```

```
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 12)
Me.ClientSize = New System.Drawing.Size(292, 266)
Me.Name = "Form1"
Me.Text = "Form1"
```

```
End Sub
```

```
#End Region
```