

Products Review



Visual Studio 2005専用
タブ機能コンポーネント

TabPlus for .NET 2.0J

精進湖計算機
立中 秀樹 TATENAKA, Hideki

Software Information

OS	Windows 2000/XP Windows Server 2003
開発環境	Visual Studio 2005
対応言語	Visual Basic C#
実行環境	.NET Framework 2.0
価格	29,400円 (ダウンロード販売のみ)

問合せ先

グレイブシティ株式会社

TEL : 022-777-8211

FAX : 022-777-8233

URL : <http://www.grapecity.com>

MAIL : sales@grapecity.com

はじめに

最近、タブ機能を持つブラウザが増えていています。Internet Explorerも次バージョン7.0ではタブ機能が使用できるようになります。個人的には単一ウィンドウ内で切り替えて使用するタイプのUIは、複数のウィンドウを表示していろいろな画面をデスクトップ上に配置できるというWindowsの利点を捨てている気がしてなりません。しかし、ひとつのアプリケーションをフルスクリーンにしてタスクバーで切り替えて使用しているエンドユーザーをよく見かけます。これを考えると、タブ機能を装備したアプリケーションは意外に有効かもしれません。

エンドユーザーから出される要求にはUI関係のものが数多くあります。今後タブブラウザが標準になってゆけば、当然ながら業務アプリケーションをタブ機能で実現したいという要望も増え

てくるでしょう。

そこで今回は、Windowsアプリケーションにタブ機能を簡単に実装できるコンポーネント「TabPlus for .NET 2.0J」(以下TabPlus)を紹介します。今回紹介するバージョン2.0JはVisual Studio 2005専用ですが、Visual Studio .NET 2002/2003用のバージョン1.0Jも販売されています。

タブ機能はVisual Studioでも標準コントロール「TabControl」として用意されていますが、TabPlusでは外観の自由度やデザイナーによる画面設計など、TabControlにはない機能が提供されています(表1)。

TabControlからの移行

TabPlusでは、Visual StudioのTabControlをTabPlusに置き換える移行機能が提供されています。そこで、まずはこの機能を用いて、TabControlを使った既存アプリケーションからTabPlus

表1: TabPlusで提供される5つのコントロール

コントロール	概要
 GCTabControl	TabPlusのメインのコントロール。20種類のスタイルとスタイルのカスタマイズ機能をはじめ、タブのキャプションや配置、背景に関する多くの設定機能を備えた拡張タブコントロール
 GCMdiTabControl	MDIフォームをタブ形式に変換して表示するコントロール
 GCLabel	文字の縦表示や均等割付など、GCTabControlが持つテキスト機能を実装した拡張ラベルコントロール
 GCContainer	GCTabControlが持つ子コントロールへのラベルと立体表示の追加機能を単独で使用可能にした拡張パネルコントロール
 GCFlowLayoutContainer	GCContainerの機能とフロー表示(子コントロールの水平/垂直方向への自動整列)機能をあわせ持つ拡張フローレイアウトパネルコントロール

言語 >>> Language

Visual Basic

ツール >>> Tool

Visual Studio 2005 Professional

を使ったアプリケーションへ移行して
みます。

移行対象は、コントロールと、プロパ
ティウィンドウで設定したプロパティ設
定のみです。コードで記述したTabCon
trolに関するイベントやメソッドは、Tab
Plus用に手動で変更する必要があります。

■ TabControlを使った アプリケーション

ここでは、Visual StudioのTabCont
rolを使ったサンプルとして図1のよう
なアプリケーションを用意しました。
ウィザードのように、ページに分けて
個人情報を登録する機能を提供します
(図2)。

このサンプルではリスト1のコードを

記述し、タブを選択するか、[<<]
[>>] ボタンでタブを移動するとTab
ControlのSelectedIndexChangedイベ
ントで移動を感知してフォームのタイ
トルを変更するようにしています。

また、3つのボタンは、タブ内ではな
くフォーム上に配置してTabControlよ
り前面に表示しています。ボタンをタ
ブ上に直接配置してしまうと、すべて

図1：TabControlを使ったサンプルの実行画面

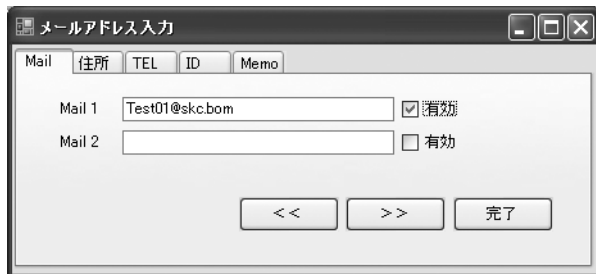


図2：サンプルのUI



リスト1：TabControlを使ったサンプルのコード

```
Private Sub TabControl1_SelectedIndexChanged(_
    ByVal sender As Object, ByVal e As System.EventArgs) _
    Handles TabControl1.SelectedIndexChanged
    Select Case TabControl1.SelectedIndex
        Case 0
            Me.Text = "メールアドレス入力"
        Case 1
            Me.Text = "住所入力"
        Case 2
            Me.Text = "TEL入力"
        Case 3
            Me.Text = "ID入力"
        Case 4
            Me.Text = "メモ入力"
    End Select
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles Button1.Click
    If TabControl1.SelectedIndex > 0 Then
        TabControl1.SelectedIndex -= 1
    End If
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles Button2.Click
    If TabControl1.SelectedIndex < 4 Then
        TabControl1.SelectedIndex += 1
    End If
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles Button3.Click
    Me.Close()
End Sub
```

