



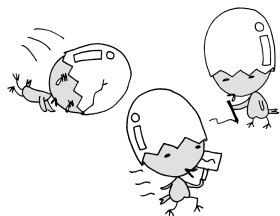
作って学ぶ
VISUAL BASIC .NET

瀬戸 遥

SETO, Haruka

<http://www.big.or.jp/~seto/>

<http://hp.vector.co.jp/authors/VA006682/>



最終回

ドラッグ&ドロップ機能を実装してみよう

Technology Tools

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

Level



Samples

・この記事で取り上げたソースコードおよびサンプルプログラムは、付録CD-ROMの¥DOTNET¥TAMAHYOディレクトリに収録しています。

¥DRAGDROP
今回のサンプル



2種類のドラッグ&ドロップ

今月は、ドラッグ&ドロップを使ったプログラムにチャレンジしてみます。

ご存知のとおり、ドラッグ&ドロップとは、マウスを使ってデータを移動して、いろいろな処理を実行する操作です。

この操作には、「アプリケーション内でのドラッグ&ドロップ」と、「アプリケーション間でのドラッグ&ドロップ」という2種類があります。

このようなドラッグ&ドロップ操作を実現するには、データを送る（ドラッグする）側と受け取る（ドロップされる）側にそれぞれ処理が必要になります。

また、ドラッグ&ドロップで受け渡すデータにはいろいろな種類があり、それを事前に指定する必要があります。

そこで、今月はテキストデータをドラッグ&ドロップする機能をプログラムに実装する基本的な方法について紹介します。



ひとつのアプリケーションに実装する

はじめに、ひとつのアプリケーション内でドラッグ&ドロップ操作を行なう方法についてまとめておきます。

ドラッグ&ドロップ機能を実装するには、データをドラッグする側と、データがドロップされる側のコントロールで次の処理を行ないます。

●データをドラッグする側

マウスイベントなどを使って、Do DragDropメソッドを実行します。このメソッドは、ドラッグ&ドロップ操作を開始するメソッドです。

引数にドラッグするデータ形式や、データをコピーするのか移動するのかなどのドラッグ&ドロップ操作の効果を指定します。

●データがドロップされる側

まず、データを受け取るコントロールの、**AllowDrop** プロパティをTrueにします。このプロパティをTrueにすると、コントロールはデータを受け入れることができるようになります。これはプロパティウィンドウで設定可能です。

DragEnter イベント

続いて、**DragEnter** イベントで、データが受け入れられる形式のものなのかをチェックするコードを記述します。

DragEnter イベントは、ユーザーがドラッグ&ドロップ操作を開始し、最初にマウスイカーソルをコントロール上にドラッグすると発生します。このイベントプロシージャの引数 **DragEventArgs** を使い、**DataObject** クラスの **GetDataPresent** メソッドで、ドロップされたデータが、指定した形式に関連付けられているかどうかを確認します。確認後、ドラッグ&ドロップ操作のターゲットのドロップ効果を設定します。

DragDrop イベント

受け入れられるデータであることを確認できたら、**DragDrop** イベントでデータを処理します。

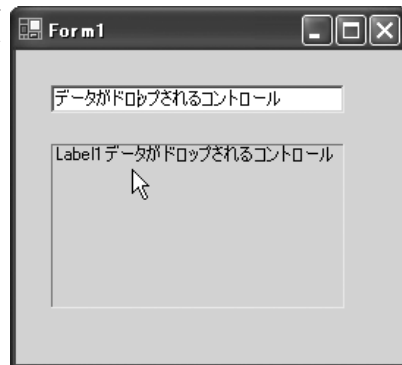
DragDrop イベントは、ドラッグ&ドロップ操作が完了したときに発生します。このイベントプロシージャの引数「**DragEventArgs**」と、**DataObject** クラスの **GetData** メソッドを使うことで、ドロップされたデータを受け取ることができます。



テキストデータの ドラッグ&ドロップ

早速、ドラッグ&ドロップ操作でテキストデータをコピーするプログラムを作ってみましょう。

図1：ドラッグ&ドロップでLabelコントロールに表示



ここで作成するのは、**TextBox** コントロールに入力されたデータを、ドラッグ&ドロップで**Label** コントロールに表示するプログラムです (図1)。

●コントロールの配置

フォームに**TextBox** コントロールと**Label** コントロールを配置します。

そして、プロパティウィンドウで**Label** コントロールの **AllowDrop** プロパティをTrueにします。これで、**Label** コントロールはデータのドロップを受け入れることができるようになります。

●ドラッグする側 (TextBox)

次に、データをドラッグする**TextBox** コントロールの処理を作成します。

ここでは、**MouseLeave** イベントを使い、**Text** プロパティに格納されているデータをすべて一括で**Label** コントロールにドロップするようにします。もちろん、プログラムの機能によっては、**MouseDown** など他のマウスイベントを使ってドラッグ処理を実行してもかまいません。

では早速、**MouseLeave** イベントプロシージャを作成してみましょう。

```
Private Sub TextBox1_MouseLeave(ByVal sender As Object, _
    ByVal e As System.EventArgs) Handles
    TextBox1.MouseLeave

End Sub
```