



特集 1

SPECIAL FEATURE

## はじめての データベース アプリケーション

2

### データバインディングを使いこなすために

# 絞り込み / 検索 / 検証 / 印刷

大澤 文孝 OSAWA, Fumitaka

## はじめに

データベースアプリケーションを作るときに厄介なのがユーザーインターフェイスです。

Windowsアプリケーションとしてデータベースアプリケーションを構築する場合には、データバインディングが基本です。

データバインディングでは、DataSet内のデータとコントロールのプロパティとのやりとりが自動化されるので、開発者は、データの取得や設定のためのコードを書く必要がありません。

しかし一方で、ユーザーは不正な値を入力することもありますから、どこ

かでデータが正当かどうかを判定する必要があります。

またデータベースに格納されているデータとは違う書式でユーザーに表示したいこともあります。その場合には、書式の変換処理を実装する必要があります。

さらに、データベースアプリケーションでは、何らかのかたちで印刷処理も実装する必要があるでしょう。

Visual Studio 2005では、フォームデザイナーや各種コントロール、そして追加されたイベントなどを使うことによって、これらの処理を、以前よりも簡単に実装できるようになりました。

本稿では、データベースアプリケーションに必要な、ユーザーインターフェイスの制御方法を、まとめて紹介します。

レベル >>> Level

1 2 3 4 5

言語 >>> Language

▪ Visual Basic

ツール >>> Tool

▪ Visual Studio 2005 Professional  
▪ SQL Server 2005 Express

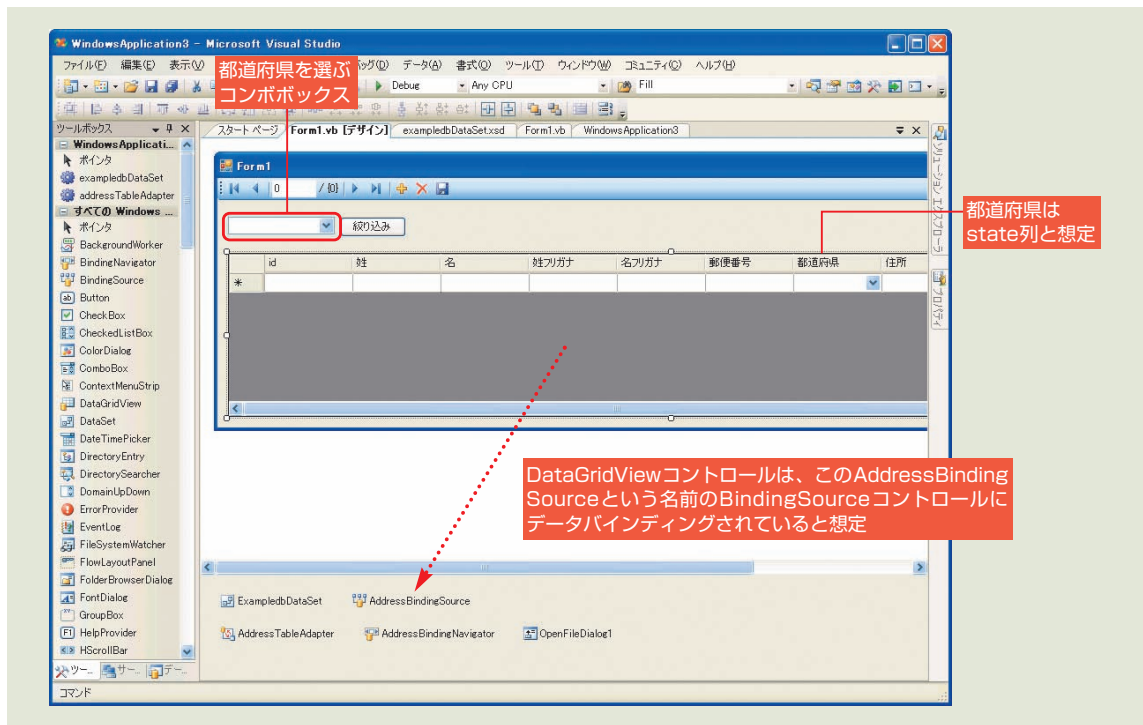
## 並べ替えと絞り込み

すでに「データベースアプリケーションの初歩の初歩」でも説明しましたが、ADO.NET 2.0では、BindingSourceコントロールを使って、カレントレコードの位置や並べ替え、絞り込みといった操作を行ないます<sup>[E1]</sup>。

# 特集 1 はじめてのデータベースアプリケーション

## SPECIAL FEATURE

図1：  
絞り込むための  
コンボボックス

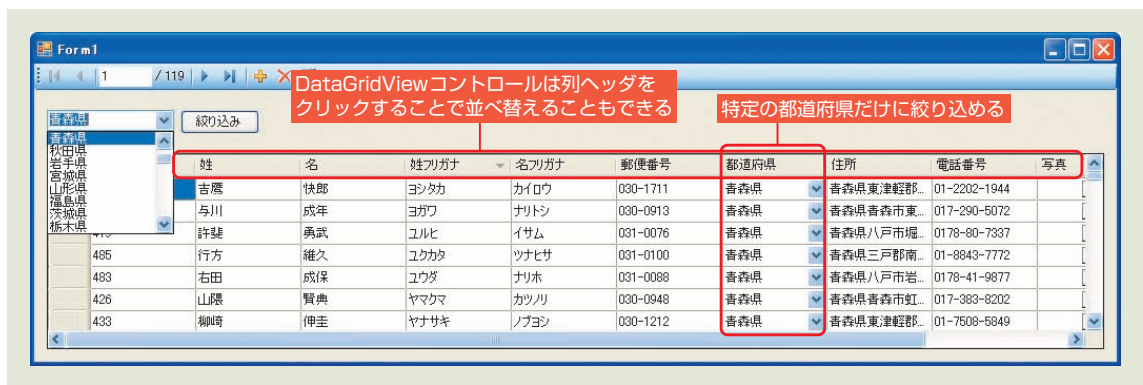


リスト1：  
Filter プロパティ  
で絞り込む

```
Private Sub Button1_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles Button1.Click
    If ComboBox1.Text = "" Then
        ' 設定解除
        AddressBindingSource.Filter = Nothing
    Else
```

```
' 設定
AddressBindingSource.Filter = _
    "state=" + ComboBox1.Text + ""
    End If
End Sub
```

図2：  
リスト1の実行結果



注1) 従来のADO.NETの開発者から見れば、BindingSourceコントロールは、DataViewコントロールとCurrencyManagerオブジェクトを合わせたものに似ています。実際、BindingSourceコントロールは、DataViewやCurrencyManagerをラッピングしているコントロールにすぎず、プロパティやメソッド、イベントは、これらを組み合わせたものになっています。もはやADO.NET 2.0では、どこにバインディングされているのかよくわからず暗黙的に作られていたCurrencyManagerを意識する必要はありません。

まずは簡単に、BindingSourceコントロールを使った、並べ替えと絞り込みから説明します。

### 絞り込みをするFilterプロパティ

絞り込みをしたいときには、条件をFilterプロパティに設定します。

たとえば、図1のようにコンボボックスを配置して、都道府県を選んだときに、その都道府県に該当するレコードだけを表示したいというような場面では、リスト1のように実装します(図2)。

リスト1にあるように、条件は、SQLのWHERE句に似た式で指定します<sup>[註2]</sup>。