

ADO.NETの初歩からXMLデータベースまで一刀両断
AccessやExcelのデータも自由自在

ADO.NETによる データベース操作のいろは

Visual Studio .NETのビジュアル開発で
簡単にデータベースアクセス

大澤 文孝
OSAWA, Fumitaka

Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2003
- ASP.NET
- Internet Information Services
- Other:
 - .NET Framework 1.1
 - Internet Explorer 6.0

Level



Samples

はじめに

.NET Frameworkでデータベースにアクセスするには、ADO.NETを使います。ADO.NETは、System.Data名前空間内に存在する、データベースにアクセスするためのクラスの総称です。

これらのクラスは、数が多く、またデータが階層構造になっているため、クラスの間関係を理解する必要があります。

そこで本稿では、SQL Serverへのデータベースアクセスを例にとり、ADO.NETを使って、どのようにデータベースを操作すべきかを説明します^[注1]。

データベースアクセスの方法は2種類

ADO.NETを使ってデータベースに

注1) SQL Serverにアクセスする場合には、System.Data.SqlClient名前空間にあるSqlXXXXのようにSqlから始まるクラスを使います。それ以外のデータベースにアクセスする場合には、System.Data.OleDb名前空間 (OleDBの場合) やSystem.Data.OracleClient名前空間 (Oracleの場合、.NET Framework 1.1以降のみ) に含まれているクラスを使います。

アクセスする場合、2通りの方法があります。

ひとつは、データベースにアクセスしたままデータベースにアクセスする「接続型データベースアクセス」、もうひとつは、データベースから一度メモリに取り込んでメモリ上で操作してあとでデータベースに書き戻す「非接続型データベースアクセス」です (表1)。

簡単に言うと、コードを記述するのは大変だけれども効率が良いのは①、たくさんのコードを書かなくても良いけれども効率が悪くなるのは②、ということになります。

今回は、Visual Studio .NETからビジュアル環境で開発できる、②の非接続型データベースアクセスを解説します。

Visual Studio .NETのビジュアル環境でデータベース開発する

Visual Studio .NETでは、SqlConnectionコントロールとSqlDataAdapterコントロールを使って、容易に開発できます。しかも「サーバーエクスプローラ」から、必要なデータベースのテー

ブルをドラッグ&ドロップすると、各種情報が設定されたSqlConnectionコントロールとSqlDataAdapterコントロールを作ってくれます [注3]。

ここでは、図1のように、SQL Serverのpubsデータベースのauthorsテーブルをドラッグ&ドロップすることになります [注4]。

すると、SqlConnection1という名前のSqlConnectionオブジェクトと、SqlDataAdapter1という名前のSqlDataAdapterオブジェクトができます。

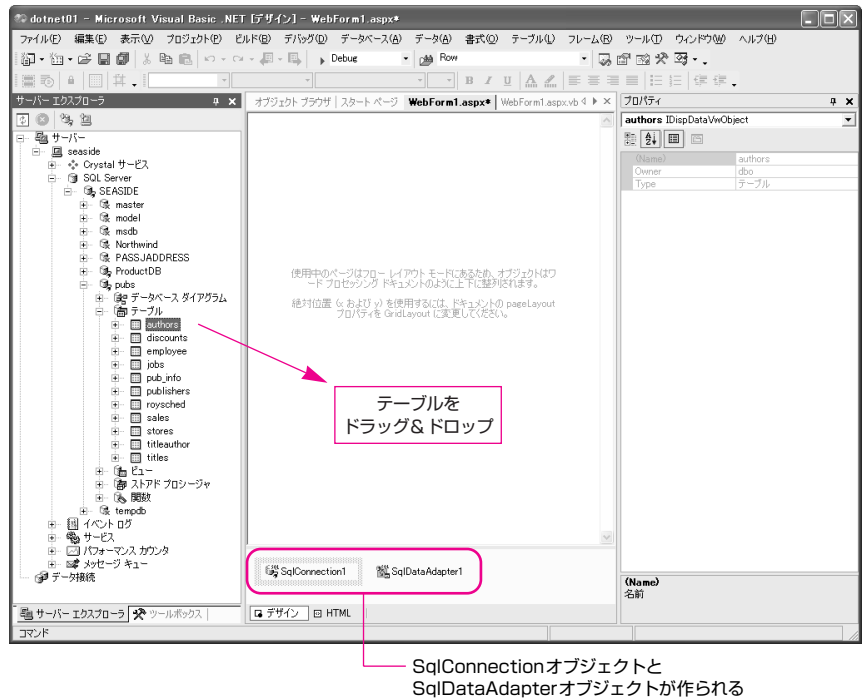
SqlConnectionオブジェクト

SqlConnectionオブジェクトは、データベースへの接続を管理するオブジェクトです。

Openメソッドを呼び出すとデータベースに接続し、Closeメソッドを呼び出

注3) サーバーエクスプローラは、[表示]メニューの[サーバーエクスプローラ]から表示できます。
注4) ここでは、ASP.NETアプリケーションを例にとって説明しますが、Windowsアプリケーションの場合にも、同様の手順で操作できます。

図1：データベースのテーブルをドラッグ&ドロップする



すとデータベースから切断する機能を持ちます。

SqlConnectionオブジェクトで、データベースの接続先を決めるのが、ConnectionStringプロパティです。

サーバーエクスプローラからドラッグ&ドロップしたときのConnection

Stringプロパティは、次のような文字列となります。

```
workstation id=サーバー名;
packet size=4096;
integrated security=SSPI;
data source=サーバー名;
persist security info=True;
initial catalog=pubs
```

表1：データベース接続の種類と特徴

接続法	①接続型データベースアクセス	②非接続型データベースアクセス
利用するクラス [注2]	SqlConnection、SqlCommand、SqlDataReader	SqlConnection、SqlCommand、SqlDataAdapter、DataSet、DataTable、DataRow
特徴	データベースに接続したままで操作します。高速なのが特徴です。SQLのクエリーを直接データベースに送信することで操作をする方式となります。Visual Studio .NETのビジュアル開発環境でのサポートはほとんどなく、自らコードを記述する開発スタイルとなります。	データベースに対して、データベースのレコードを取得するクエリー（SELECT文など）を実行し、メモリ上に読み込みます。この時点で、データベースとの接続は断ち切られます。そして、読み込んだメモリ上のデータを変更し、その後、データベースに再接続して、更新のあったレコードに関してデータベースに対して更新するクエリー（INSERT文、UPDATE文、DELETE文など）を実行し、一気に更新をかけます。簡単に言えば、レコードをいちどメモリ上にコレクションとして取得し、その内容を変更したあとにデータベースに書き戻す方式です。たくさんのレコードを読み込む場合には、たくさんのメモリを消費するという欠点があります。しかしメモリ上のデータをコレクションとして扱えるため、データの参照や加工が行ないやすいという特徴があります。Visual Studio .NETには、SqlDataAdapterコントロールやDataSetコントロールなどの各種コントロール、そしてウィザード機能などが提供されており、ビジュアル開発環境で、コードの自動生成ができます。

注2) 利用するクラスとしては、データベースアクセスに最低限必要となるものしか記述していません。