

データベースの更新がしやすくなったDataTable

# DBアプリケーション構築を楽しくする ADO.NET 2.0

大澤 文孝  
OSAWA, Fumitaka

Level

1 2 3 4 5

Technology Tools

- Visual Basic
- Visual C#
- Visual C++
- SQL Server
- Oracle
- Access
- ASP.NET
- Other:
  - Visual Studio 2005 ベータ2
  - SQL Server 2005 ベータ2
  - ADO.NET 2.0

Samples



はじめに

Visual Studio 2005では、「ADO.NET 2.0」を使ってデータベースにアクセスします。

ADO.NET 2.0の基本的な考え方は、従来のADO.NET 1.xと変わりません。

しかし、DataTableオブジェクトでは、DataReaderオブジェクト<sup>[注1]</sup>からデータを直接読み取れるように改良されていたり、Webアプリケーションでは、開発者がコードを書かなくてもデータベースへの書き戻しができるようになっていたりするなど、細かいながらも開発しやすくする工夫が随所に見受けられます。

またADO.NET 2.0は、SQL Server 2005との親和性も高まりました。SQL

Server 2005でサポートされた新しい型のサポートはもちろん、SQL Server 2005上でデータが更新されたときに、そのクエリ通知をイベントとして受け取る機能も備えています。

ADO.NET 2.0の新機能は表1にあげたように数多く、とてもここですべてを紹介することはできません。

そこで本稿では、ADO.NET 2.0の新機能のうち、開発にあたって知っておきたい基礎事項と、実際のアプリケーションで使い道が多そうな機能に絞って説明します。

なお、SQL Server 2005に限った機能についてはそれぞれの内容が深く、扱いきれないため、それらについては、また機会を改めて説明したいと思います<sup>[注2]</sup>。

注1) SQL ServerならSqlDataReaderオブジェクト。以下同様に、データプロバイダごとの接頭辞は省略。  
注2) SQL Server 2005との連携機能を説明するなら、ADO.NET 2.0の新機能だけでなく、ストアプロシージャを.NETで開発できるようになったという点にも触れないとならないでしょう。

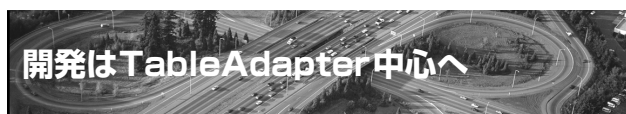
表1：Visual Studio 2005におけるデータベースアクセスの主な新機能

## 汎用的な新機能

機能	概要
TableAdapterを使った開発	Windows フォームでは、従来のDataSetコントロールやDataAdapterコントロール、Connectionコントロールの代わりに、TableAdapterという自動生成されるクラスを使って開発するようになった
カレントレコード位置を操作するBindingSourceコントロールの提供	Windows フォームでは、BindingSourceコントロールを使うことで、カレントレコードの移動操作がしやすくなった
Webアプリケーションにおけるデータ更新のサポート	Webアプリケーションにおけるデータバインディングは、データベースからの読み取りだけでなく、データベースへの更新操作もサポートされた
プロバイダファクトリを使ったデータプロバイダに依存しないオブジェクトの作成	DbProviderFactoryクラスを使って、アプリケーション構成ファイル内に記述したデータベース接続情報から、汎用的なコードでConnectionオブジェクト、Commandオブジェクトなどを作成できるようになった
DataTable (DataSet) と DataReaderとの連携	DataAdapterオブジェクトを使わなくても、DataReaderオブジェクトから直接DataTable (DataSet) にレコード情報を読み取れるようになった。またDataTableオブジェクトから1レコードずつ読み込むためのDataTableReaderオブジェクトがサポートされた
スキーマ情報の取得	ConnectionオブジェクトのGetSchemaメソッドを用いることで、データベースやテーブルのスキーマ情報を取得できるようになった
データベース接続のプーリング動作の設定	データベース接続のプーリングをメソッドで制御できるようになった (現状はSqlClientデータプロバイダとOracleClientデータプロバイダのみ)
非同期でのSQLクエリの実行	クエリの結果を待つことなく、非同期でクエリを次々と送信できるようになった。たとえば、ExecuteReaderメソッドには、非同期での操作に対応するBeginExecuteReaderメソッド (非同期クエリの開始)、EndExecuteReaderメソッド (非同期クエリの終了、結果読み取り) といったように、BeginXXX、EndXXXのメソッドが用意された (現状はSqlClientデータプロバイダのみ)
統計情報の取得	SqlConnectionオブジェクトのStatisticsEnabledプロパティをTrueにすると、データプロバイダの統計情報をリアルタイムに取得できるようになった (現状はSqlClientデータプロバイダのみ)
データの一括転送	SqlBulkCopyクラスを使って、テーブル内容を別のデータベースに一括転送できるようになった (SQL Serverのみ)

## SQL Server 2005に固有の機能のサポート

機能	概要
MARS (Multiple Active Result Sets) のサポート	SQL Server 2005ではカーソルレス結果セットを複数保持できるようになった。ADO.NET 2.0はこれに対応し、ひとつのコネクション (SqlConnectionオブジェクト) で同時に複数のクエリを実行することができるようになった。言い換えれば、ひとつのSqlConnectionオブジェクトを閉じることなく、ExecuteReaderメソッドを複数呼び出し出してもかまわないことになった
クエリ通知のサポート	SQL Server 2005のクエリ通知に対応し、通知が起きたときに、イベントとしてその情報を受け取れるようになった。具体的には、SqlNotificationRequestクラスとSqlDependencyクラスが提供された
新しいデータ型のサポートなど	その他、SQL Server 2005に特有の型やトランザクションレベル、クライアントフェイルオーバーなどの機能がサポートされた



## 開発はTableAdapter中心へ

Visual Studio 2005では、データベース接続の構成方法など、統合開発環境における操作も大きく異なっています。

そこでADO.NET 2.0の機能説明に入る前に、Visual Studio 2005におけるデータベース操作について説明しておきます。

従来のVisual Studio .NET 2003でWindows フォームを構成するときには、Connectionコントロールを使ってデータベースとのコネクションを構成し、DataAdapterコントロールを通じてDataSetコントロールに流し込むという方法をとっ

ていました。

それに対して、Visual Studio 2005のWindows フォームでは、TableAdapterを使ってデータベース操作をします。

TableAdapterは、従来のConnectionコントロールとDataAdapterコントロールを合体した機能を備えたものです。

## TableAdapterはウィザードで作る

TableAdapterは、ADO.NET 2.0の機能ではなく、Visual Studio 2005のウィザードによって自動生成されるクラスです<sup>[注3]</sup>。

TableAdapterは、「データソース構成ウィザード」を使っ