

# Accessシステム 超リフォーム術

ポイントを抑えて楽々リフォーム

第6回

## MDBからAccessプロジェクトへの リフォーム

T'sWare

星野 努 HOSHINO, Tsutomu

<http://homepage1.nifty.com/tsware>



### Technology Tools

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

### Level



### Samples

### はじめに

Access 95や97の頃でも、SQL ServerなどのサーバーサイドのRDBMSを利用することは可能でした。しかし、それは、ODBCという、今となっては古い技術を通してデータベース上のデータを単に「利用」できるに過ぎません。SQL Server上にテーブルを作るにもビューやストアドプロシージャを編集するにも、SQL Serverの管理ツールを使い、その完成品をAccessから呼び出すというような、2元的な開発が必要だったのです。

しかし、Access 2000から登場したAccessプロジェクト (ADP) は、その形態を一変させました。SQL ServerやMSDEのデータを利用できるのは当然のこと、それらのオブジェクトを作ったり編集したりといった「生成/管理」までをAccessで担当できるようになったのです。Accessがユーザー用アプリケーションであるとともに、SQL Serverなどの開発用クライアントツールに

発展したといえます。

そこで今回は、これまで行ってきたアップサイジングの総仕上げとして、Accessプロジェクトの概要とともに、既存のAccessデータベース (.MDB) をAccessプロジェクトへと移行する方法について説明します。

### Accessプロジェクトとは?

#### 🏠 MDBの問題点を解決

Accessデータベース (以下MDB) をC/Sベースで利用するには、次のような問題点がありました。

**問題点 1** MSDEのテーブルやビューを“リンクテーブル”として扱うことができない

**問題点 2** ストアドプロシージャを利用できるのは“モジュール”からだけである

**問題点 3** MSDEのオブジェクトを直接生成したりメンテナンスしたりす

ることができない

**問題点 4** JetデータベースエンジンおよびODBCを経由するため、パフォーマンスが十分でない

これらの問題を一気に解決してくれるのがAccessプロジェクト（以下ADP）です。

ADPは、Jetデータベースエンジンを一切使わず、OLE DBを介してMSDEにダイレクトにアクセスすることができます。さらに、MSDE上のオブジェクト自体を直接的にコントロールできる、C/S環境でのAccessクライアント用ファイル形式です。

## ADPの外観と特徴

次に、Accessのデータベースウィンドウの外観を通して、ADPの特徴を見てください（図1）。左側が従来からのMDBのデータベースウィンドウ、右側がADPのデータベースウィンドウです。

この2つのオブジェクトバーを見比べてみると、「ADPの“データベースダイアグラム”<sup>注1</sup>を除いて両者はまったく同じ」ということに気付くはずですが。

注1) データベースダイアグラムは、MDBの“リレーションシップ”と同等のものです。

注2) ただし、前号で説明したように、ADPでは、モジュールからテーブルなどにアクセスするにはDAOではなく「ADO」を使わなければなりません。このように、オブジェクトの種類は共通でも、その中味すべてがまったく同じというわけではありません。

注3) ADPでは、特定のデータベースサーバーかつデータベースのみが対象となります。多くのサーバーやデータベースを管理したり、より高度なサーバー側の開発/管理を行なったりする場合には、やはり専用のツールのほうが充実した機能を備えています。

図1：MDBとADPのデータベースウィンドウ

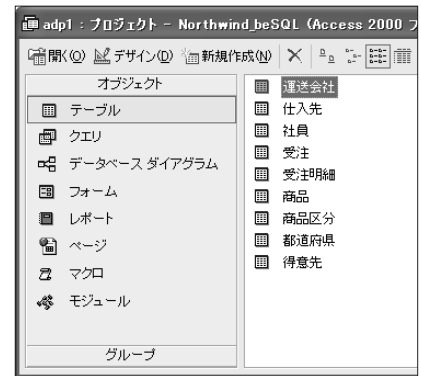
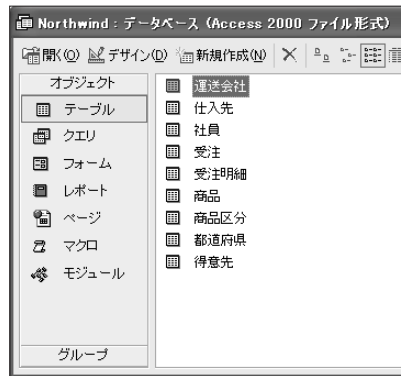
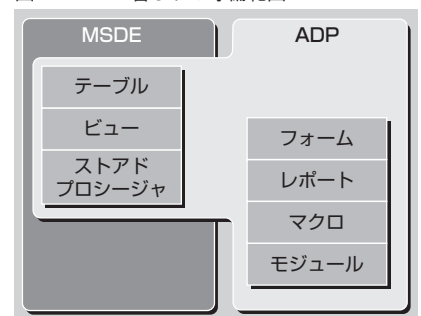


図2：ADPに含まれる守備範囲



ただし実際には、ADPの「テーブル」の中味はMSDE上のテーブルであり、ADPの「クエリ」はMSDE上のビューやストアドプロシージャなどとなります。そして、これらは、ADPの次のような特徴を示しているといえます。

**特徴 1** MSDE上のオブジェクトを、デザインも含めて直接表示させることができる

**特徴 2** フォーム、レポート、モジュールなどは、ADPでもMDBでも同じように扱うことができる<sup>注2</sup>

つまり、ADPはMSDEに密接したファイル形式でありながらも、MSDEや

SQL Serverの管理ツール「Enterprise Manager」のようなサーバー側の開発/管理だけでなく<sup>注3</sup>、従来通りの方法で、フォームやレポートなどのクライアント側オブジェクトも扱えるという、広い守備範囲を持った形式といえるでしょう（図2）。

## Column

### ADPのデメリット

MDBに慣れた開発者やシステム管理者には以下のような面倒な面もあります。

- ・ Accessだけ起動してもADPファイルのオブジェクトの中までアクセスすることができない。常にMSDEを起動しておくことが必要
- ・ MSDEにアクセスできるLAN内であれば基本的には開発したADPファイルを配布するだけで済むが、ネット公開や販売など、広く配布する場合には常にADPとMSDEをセットで配布する必要がある。当然のことながら、ユーザー側では、サーバー側とクライアント側両方のセットアップを考慮しなければならない