# ASP.NET INSIDE

ASP.NETをめぐる思考と試行とその冒険

第15回

# WebアプリケーションとWindowsアプリケーションのコードを共通化する―その4―

## 2種類のXML Webサービスを使ったデータの更新

秋月 巌 *AKIZUKI, Iwao*
秋月巌ソリューション事務所

**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\ASP_NETディレクトリに収録しています。

**\SAMPLE**
サンプルプログラムとソースコード

### .NET 時代が変われば、必要なものも変わる

今回の記事とは何の関係もないのだが、最近、私はDHTMLに注目している。DHTMLがそこそこ華々しくデビューしたとき、私も一通りの評価作業は行なった。そのときに私が出した結論は、「どうせこんなにInternet Explorerに依存しているならば、ActiveXコントロールのほうが柔軟性が高いし、信用もできる。それに、いずれJavaアプレットも実用的になるだろう」というものだった。それからひとつの時代が過ぎ、パーソナルコンピュータ用のWebアプリケーションならば、まぁInternet Explorerだけで動けば、互換性は何とか合格という時代になった。そのプロセスでクライアントJavaの夢は潰え、Webアプリケーションはどちらかというとクライアントの処理に依存しない方向のまま進化を止めている。

その一方でDHTMLはInternet Explorer 4.0以来、ほとんど変化せず、ある意味枯れた技術になりつつある。私の経験からいうと、クライアントスクリプトはWebブラウザのバージョンにかなり依存する。最近、評価作業をしたASP.NETのWebフォーム用のサーバーコンポーネントが、機能のかなりの部分をDHTMLに依存しているのを見て、これでWebブラウザをバージョンアップしても平気なのだろうかと心配してしまう。しかし、DHTMLの進歩の止まり具合をみれば、それは余計な心配なのかもしれない。

本誌編集長がマイクロソフト株式会社（Microsoftの日本法人）の人と話をしたときにDHTMLに話題をふると「もうレガシーシステムでしょ」と言われたという。レガシーシステムと聞いて、過去の古びたテクノロジーととらえるか、あるいは安定した信頼できるシステムととらえるかは個人差があるだろう。すでに過去のものとなったシステムを採用する最大のリスクは、

そのテクノロジーがあるポイントでサポートされなくなることである。ただ、DHTMLを使用したサイトがわりと多い（Microsoftが運営するHotmailもそうである）ことを考えれば、このままサポートをしなくなるということは考えにくい。

## .NET いまだからこそDHTML（かも）

クライアントコンポーネントを使用せず、DHTMLで高度なインターフェイスを備えたWebアプリケーションというのは、今日だからこそ検討してもいいのかもしれない。たとえば、DHTMLを使ったワープロソフトや表計算ソフトは現実的だろうか。問題は作成した文書を保存するのに、データを何かしらの形に変えてアップロードしなければいけないということだろう。今回の記事のテーマでもあるのだが、Webアプリケーションのデータアップロード機能は、かなり貧弱である。とはいえ、考えてみればDHTMLはクライアントのファイルのロード機能がないので、プログラムと同時に文書データを（ファイルとしてではなく）スクリプト埋め込みのデータとしてダウンロードしなければならない。ダウンロードはともかく、ある程度大きいデータをクライアントで実行されたアプリケーションにロードするのは、クライアントスクリプトの堅牢性では重荷かもしれない。

まぁ、そこまでいかないにしても、DHTMLを要所で使用することは有効かもしれない。たとえば、次回、コンポーネントを使わないでグリッド形式でデータを更新するWebアプリケーションを紹介するが、このグリッドは矢印キーの操作に対応していない。つまり、カーソルを矢印キーで移動することができない。これはHTMLの限界からくる制約である。しかし、DHTMLを利用すれば、実現が可能である。とはいえ、DHTMLの利用がある意味での賭けであることは間違いない。



## .NET データ更新も含めて2つのXML Webサービスを比較する

さて「WebアプリケーションとWindowsアプリケーションのコードを共通化する」シリーズもついに4回目となった。今回は、ようやく今まで提案してきた方法を利用して、データの更新を行なう。最初にaspxファイルに記述されたASP.NETプログラムをWebClientクラスから呼び出す方法（以下aspx Webサービス）を紹介し、次にMicrosoftのWebサービスであるXML Webサービス（asmxファイル）を使用する方法を紹介する。

この2つを比較して、少なくともここで紹介するようなテキストベースでデータベースにアクセスする上で、「XML Webサービスのもつ利点はない」と判断し、以降はaspx Webサービスでのサンプルのみ紹介してゆく。XML Webサービスの問題点は先月号でも説明したように、「参照設定と、その変更方法のわずらわしさ」である。

## .NET サンプルプログラムのインストールと実行方法

### ■ 実行環境について

今回提供するサンプルプログラムもInternet Information Services（IIS）、.NET Framework、SQL Server（またはMSDE）がインストールされていることが前提となっている。SQL Serverにはsaログインがパスワードなしでアクセスできる必要がある（開発環境とはいえ、この設定はMicrosoftも推奨しないし、私もお勧めしない）。パスワードが設定されている場合には、各プログラムの接続文字列を変更する必要がある。また、標準でインストールされる「pubs」データベースを使用するので、それが利用可能な状態である必要がある。

### ■ aspxファイルのインストールと実行方法

付録CD-ROMに収録しているSAMPLE\aspxsamples2フォルダをWebで公開されているフォルダのルートフォルダにコピーする。そしてWebブラウザで、次のよう