

Windows プログラミング

第 6 回

ネットワーク／インターネット関連のライブラリ

こだか かおる
KODAKA, Kaoru

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
Express Edition April CTP

Samples

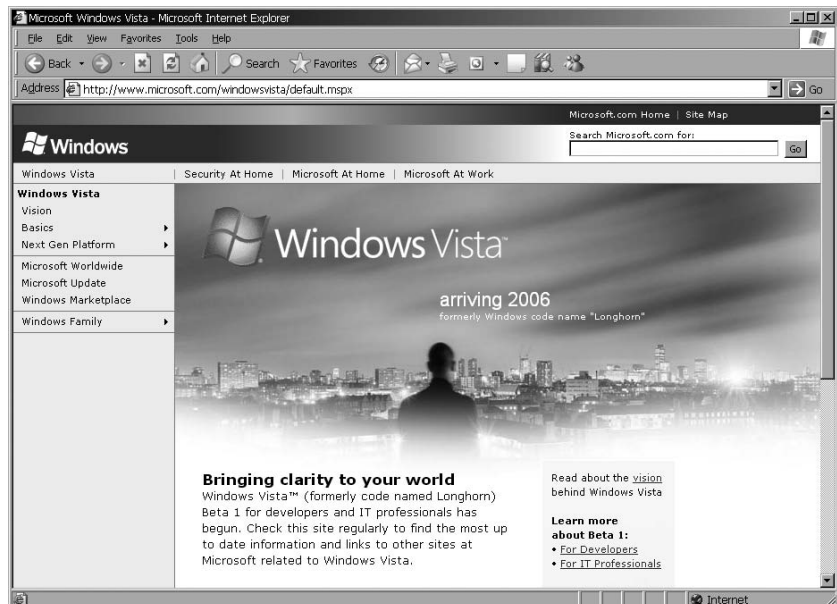
次期Windowsの ベータ版

最近の大ニュースといえば、7月の末にWindows Vistaのベータ1が提供されたことでしょうか(図1)。Windows Vistaは、それまで“Longhorn”というコードネームで呼ばれていた、Windows XPの後継となる最新のWindowsです。

最初にその名前を聞いたときは、「どっかで聞いた名前……そんな車があったな」という感想でした。海外のWebサイトでも、「しっくりこない」「変な名前だ」といった評価があるようですが、Windows XPがそうであったように、いずれ慣れていくことでしょう。

MSDN Subscriber Downloadでの提供が始まったものの、サイズが2Gを超

図1: Windows Vista ベータ1登場!



える非常に大きいものとなっています。まだ、私はWindows Vistaを試していませんが、周りの人の話を聴いたり、インストールレポートを読んだりしてみると、なかなかよさそうな印象です。入手/インストールでき次第、いろいろと試してみたいと思います。Avalon、WinFSに代表される新機能に注目が集まっていますが、開発者としては.NET Framework周りもとても気になるところです。機会があれば、この連載でもWindows Vistaにおける.NET Frameworkについて、じっくりと取り上げてみたいと思います。

ネットワーク/インターネット関連の新機能

.NET Frameworkはインターネット時代の開発フレームワークということで、数多くのネットワーク/インターネット関連の機能がクラスライブラリとして用意されています。とはいえ、ひと括りに「インターネット」といっても、その範囲はとても広く、また、日進月歩で進化する世界ですから、到底、すべてを網羅することはできないでしょう。そうなると、足りない機能は自分で実装したり、すでにどこかで誰かが実装したライブラリを探すことになります。

今回紹介する新機能も、すでにサードパーティなどからライブラリとして提供されているのを見たことがあるかもしれません。ようやく.NET Frameworkに搭載されたという感がありますが、やはり本家である.NET Frameworkに含まれるのはうれしいものです。自分で

実装したライブラリには、どんな不具合が隠れているかわかりません。その点、.NET Frameworkに含まれていれば（もちろん不具合が含まれている可能性はあるものの）、安心して使うことができます。

FTP

FTP (File Transfer Protocol) は、ご存知の通りファイルを転送するためのプロトコルです。WebサイトにHTMLファイルをアップロードするときなどに使われる、極めて一般的なプロトコルなので、使ったことがあるという人も多いと思います。たいていは、専用のFTPクライアントを使ってアクセスすることになります。Windowsでは、コマンドラインツールのftp.exeが用意されています。また、Internet ExplorerでもFTPサーバーに接続することが可能です。

.NET Framework 2.0には2種類のFTP機能が用意されています。ひとつはWebClientクラス、もうひとつはFtpWebRequestクラスです。WebClientと

いう名前から、ついついHTTP専用かと思ってしまうのですが、FTPについても機能を提供するようになりました。

◎ WebClientを利用したFTP処理

まず、WebClientクラスによるFTP処理について見ていくことにしましょう（図2・リスト1）。

実際のところ、WebClientを利用したFTP処理は、とても簡単です。まず、ユーザー名とパスワードをNetworkCredentialに設定します。そのNetworkCredentialは、そのままWebClientオブジェクトのCredentialsプロパティに設定します。あとは、FTPサーバー側のファイル名、ローカルコンピュータ側のファイル名を指定して、DownloadFile()、あるいはUploadFile()メソッドを呼び出すだけです。

このとき、ダウンロード、あるいはアップロードするファイルは、フルパスで指定しなくてはなりません。ひとつだけ注意する点としては、この方法ではアップロード、あるいはダウンロードが同期的な処理となります。対象のファイルが小さい、または、バッチ処理で行なうような場合はそれでもよ

図2：WebClientサンプル

