

業務アプリケーションにも 生かせるノウハウ

設定情報の保存と取得

初音 玲
HATSUNE, Akira

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¥F01_04ディレクトリに収録しています。

¥REGRW
レジストリ設定サンプル
¥INIRW
INIファイル設定サンプル
¥XMLRW
XMLファイル設定サンプル

設定情報とは

業務アプリケーションを作成していると、ちょっとした情報を保存しておいて利用するような仕組みが必要になるときがある。ネットワークで接続されたRDBMSに設定情報を格納して、どのPCでログオンしても自分用の設定情報を使えるというのもよいが、たとえば、

情報を格納したRDBMSを次回から初期表示する

といった利用は不可能だ。そのためRDBMSに接続する前に必要になる設定情報は、PCに設定情報を保存しておくのがよい。もちろん、この他のケースでも、RDBMSに格納するまでもない情報というのも存在するだろう。

本稿では、業務アプリケーションを作成する上で、使う頻度が高いと思われる「設定情報の保存と取得」について、Visual Basic .NETのサンプルで解説すると共に、そこで得られるノウハウを紹介してゆきたい。

レジストリに保存する方法

レジストリに設定情報を保存する利点は、「Windowsにログオンしているユーザーごとの設定をプログラム側で意識せずに行なえる」という点だ。

なぜ、そのようなことが可能かといえば、レジストリには、ユーザーごとの領域があるからだ(図1)。このようにレジストリ[HKEY_USERS]の配下は、ユーザーごとに別々の領域が割り当てられている。

ユーザーごとのレジストリの指定方法

レジストリ[HKEY_USERS]配下は、[S-1-5-18]のようにユーザーを識別するための識別子ごとにレジストリが分かれている。どのユーザーがどの識別子であるかは一目ではわからない。

それでは、どのように現在Windowsにログオンしているユーザー用のレジストリを指定したらよいのだろうか。

その答えは、レジストリ[HKEY_CURRENT_USER]が握っている。現

在Windowsにログオン中のユーザーのレジストリを扱うときには、[HKEY_CURRENT_USER]を指定すれば、自動的に、[HKEY_USERS]配下の該当ユーザー用レジストリを指定したことになるのである。

レジストリの取り扱い

Visual Basic .NETでレジストリに情報を保存したり取得したりする方法を調べてみると、保存にはSaveSettingステートメント、取得にはGetSetting関数が用意されていることがわかる。

さらに、この2つの機能を調べてみると、利用できるレジストリは、[HKEY_CURRENT_USER] - [Software] - [VB and VBA Program Settings]配下の領域に限られている。

そして、その領域のレジストリに対しては、SaveSettingステートメントで保存し、GetSetting関数で取得できるので、Windowsにログオンしているユーザーごとの設定情報の保存と取得はすべて同じコードで実現できる。

図1：レジストリ [HKEY_USERS]

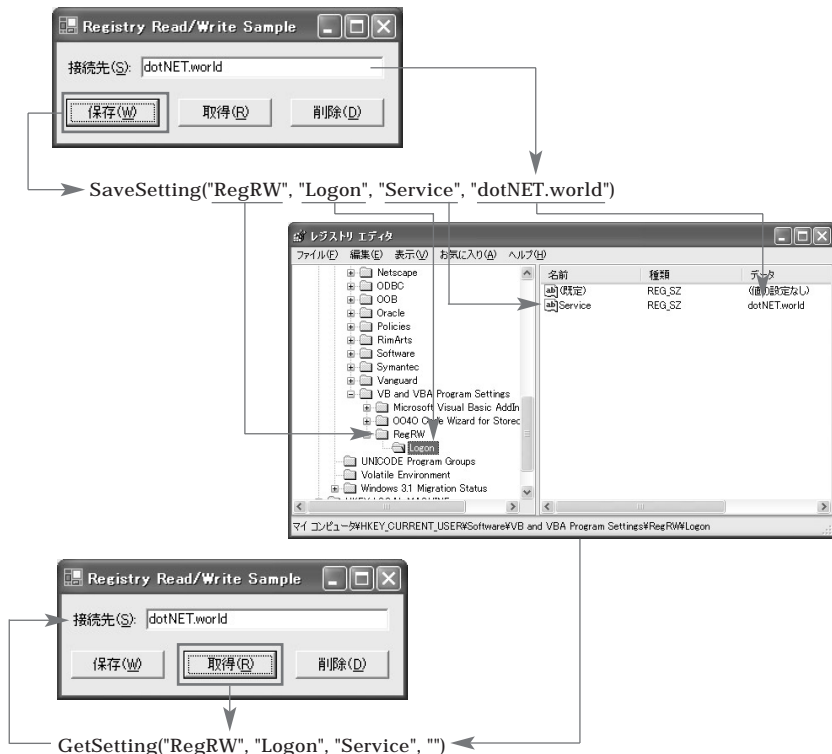


レジストリで設定情報を保存 / 取得 ~ サンプル RegRW

ここまでで、やりたいこと（設定情報を保存 / 取得したい）が明確になり、実現する方法（レジストリを使う）も決定した。そこで早速、ユーザーごとに設定情報を保存 / 取得するサンプルを作成してみよう。

まず、設定情報の流れを見てみると、図2のようになる。ここでは、SaveSettingステートメントとGetSetting関数の引数の対応を頭に入れておくと、実際にコード（リスト1）を記述するときに間違いが少ない。

図2：レジストリに設定する



SaveSettingステートメント
(リスト1-)

SaveSettingステートメントの引数は、

- AppName：アプリケーション名
- Section：セクション名
- Key：キー設定名
- Setting：設定値

という構成になっている。

ヘルプによると第1パラメータの「AppName」は、「アプリケーション名またはプロジェクト名を含む文字列」と説明があるが、アプリケーション名やプロジェクト名を含んでいない文字列を指定してもエラーにはならない。

レジストリの競合を避けるためには、次のようにするのがよいと思われる。