

Programming for Windows CE W-ZERO3用 アプリケーションの 作り方

Visual Studio 2005を使った
モバイルアプリケーション

2 アプリケーション作成の実際

これであなかもW-ZERO3アプリケーションを
構築できる

立中 秀樹
TATENAKA, Hideki

特集

はじめに

Visual Studio 2003が登場した際、これまでのeMBEDDED Visual Basicに変わって.NET Compact Framework 1.0 (以下CF1.0)が発表され、Windows CE用アプリケーションは.NET世代へと移り変わりました。

Compact Frameworkは、その名のとおりWindows CE OS上で動作する

.NET Frameworkのサブセット版で、クラス名やメソッドなどが.NET Frameworkと共通化され、デスクトップアプリケーション開発者が違和感なくデバイスアプリケーションを構築できる“はず”でした。

というのも、クラスのメソッドがなかったり、そもそもクラス自体が存在しなかったりと、.NET Frameworkの豊富な機能に親しんでいる開発者からすると、思った通りのアプリケーシ

ョン作成はなかなか多難だったのです。

しかし、昨年発表された.NET Compact Framework 2.0 (以下CF2.0)では、大幅なバージョンアップが施され、機能的にもデスクトップアプリケーションに迫るアプリケーションの作成が可能になりました。

たとえば、Microsoftのサイトで『Visual Basic .NETを使用してFTPサイトにアクセスする方法 (<http://support.microsoft.com/default.aspx?scid=kb;ja;832679>)』として公開されているコードでは、ほんの数か所コードを修正するだけで同様の機能をWindows CEアプリケーションに付加することができます。

さらに、Windows Mobile 5.0 (以下Windows CE 5.0)では新たにDirectXや内蔵カメラ、Pocket OutlookにアクセスするためのAPIも提供され、単にデスクトップアプリケーション開発との差異をなくしただけではなく、Windows CEデバイスの特性を生かしたアプリケーションの構築が可能になっています。

本稿ではCF2.0の新機能を用いてサンプルアプリケーションを作成しながらVisual Studio 2005とCF2.0を用いた基本的な開発手法を紹介します。

開発の全体像

本稿では最新のWindows CE 5.0を採用したデバイスで、もっとも一般的なPDAを対象とします。

Windows CE 5.0用のアプリケーションの開発を行なうには以下のものが必要になります。

レベル >>> Level

1 2 3 4 5

サンプル >>> Sample

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、<http://www.shoeisha.com/mag/windev/>からダウンロード可能です。

ツール >>> Tool

- Visual Studio 2005 Professional
- Windows Mobile Version 5.0 SDK for Pocket PC
- ActiveSync 4.1
- 実機 (W-ZERO3)

言語 >>> Language

- Visual Basic

- Windows PC (開発を行なうPC)
- Visual Studio 2005 Standard Edition以上^[注1]
- Windows Mobile Version 5.0 SDK for Pocket PC (含むエミュレータ)^[注2]
- ActiveSync4.1 (VisualStudioとデバイスの接続に必要)
- 実機となるデバイス^[注3]

それでは、簡単なアプリケーションを作成しながら、開発のようすを説明してゆくことにしましょう。



VS2005の起動とテンプレートの選択

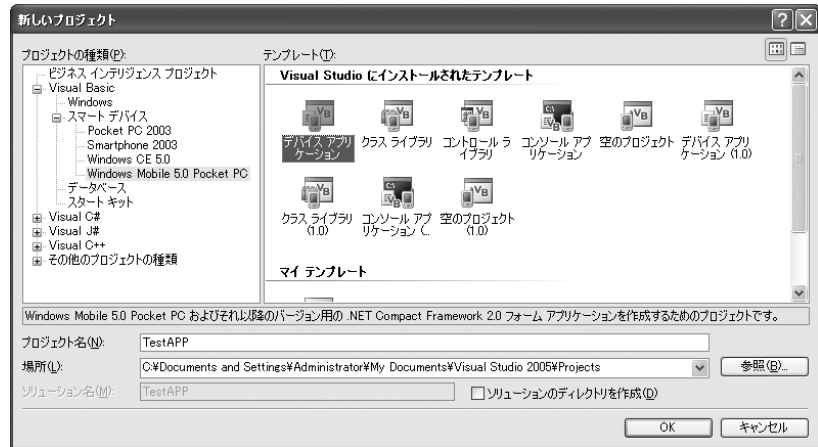
Visual Studioを立ち上げ、ファイルメニューから「新規作成」-「プロジェクト」をクリックし、「新しいプロジェクト」ダイアログを開きます。

Pocket PCのSDK (Windows Mobile Version 5.0 SDK for Pocket PC) がインストールされていればプロジェクトの種類に「スマートデバイス」が追加され、その下に「Windows Mobile 5.0 Pocket PC」が追加されます(図1)。

テンプレートペインには、通常のWindowsアプリケーション作成時と同じように、クラスライブラリやコントロールライブラリ用のテンプレートも用意されています。.NET以前の開発環境に比べて、より充実しているのがわかりますね^[注4]。

通常アプリケーションを作成する場合は、「デバイスアプリケーション」を選択します。プロジェクト名を入力して[OK] ボタンをクリックすれば、Form1が追加されたプロジェクトが作成され、IDEに読み込まれて開発を開始できま

図1：SDK導入後のVS2005



す。

ここではプロジェクト名に「TestAPP」と名づけました。

フォームの設定

新規にプロジェクトを作成すると、初期画面にはデフォルトのForm1という名称のフォームが表示されます。通常のWindowsアプリケーションと異なり、フォームの周りにはスキンが表示され、Pocket PCの実機上に表示されているように見えます(図2)。

このスキンは、Formの「Skin」プロパティを「False」にすることで、非表示に設定することもできます。

後は通常のWindowsアプリケーションの開発と同様、フォームにコントロールを配置してアプリケーションを作成してゆくのですが、その前に、このフォームに対して行なわなければならない設定があります。

先にWindows CEには複数のバージ

ョンがあると説明しましたが、開発に際して選択したPocket PCの中にも、機能や画面サイズによって、さらにいくつかの種類が存在します。

たとえば、本稿で使用しているWindows Mobile Version 5.0 SDK for Pocket PCには、利用可能なフォームファクターとして、表1に示した種類があります。

ここではVGA表示可能なPocket PCを対象とするため、Form1の「FormFactor」プロパティで「Windows Mobile 5.0 Pocket PC VGA」を選択します。

コントロールとメニューの追加

表示されたフォームにコントロールを貼り付け、イベントのコードを記述する開発スタイルは、Windowsアプリケーションとまったく同じです。

ここでは、

- Labelコントロール (2つ)

注1) SDKのみでコマンドラインからビルドすることも可能ですが、現実的ではありません。

注2) Windows CEには複数のバージョンやモデルが存在するためターゲットとなるデバイスごとのSDKのインストールが必要です。

注3) エミュレーション上での開発も可能ですが実機上でなくては動作しない場合もあります。

注4) 「(1.0)」と追記されているテンプレートは、CF1.0用アプリケーション作成用です。実際に試してみましたが、CF2.0用と混同してしまったり、プロジェクト設定などに違いがあったりするため、CF1.0用の開発を行なう場合はVisual Studio .NET 2003上で行なったほうが確実かもしれません。