

# 切れ味の鋭さが光る 玄人ツールVC++

●コンパイラ／開発環境／ツール

田中 達彦 TANAKA, Tatsuhiko

## はじめに

Visual C++ 2005は、ほぼ2年半ぶりにリリースされるVisual C++の最新バージョンです。Visual Basic 2005やVisual C# 2005に関する情報は数多く

提供されていますが、Visual C++ 2005についてはそれほど多くの情報が提供されているわけではありません。だからといって新しく搭載された機能が無いわけではありません。むしろ、Visual C++ 2005はVisual BasicやVisual C#以上にダイナミックに進化しているのです。

Visual C++ .NET 2002や2003では.NET Framework対応のアプリケーションを作成するためにC++マネージ拡張という機能を搭載していました。Visual C++ 2005では、.NET Frameworkに完全に対応するために「C++/CLI」という機能を搭載しています。C++マネージ拡張では.NET Frameworkの機能を使用できるというレベルでしたが、C++/CLIでは.NET Framework完全準拠のアプリケーションを作成するレベルになっています。特に、Windows Vista用のアプリケーションの開発を見据えた場合、C++/CLIは不可欠なものであると言えます。

.NET Frameworkに対応したアプリケーションを作成しない場合でも、Visual C++ 2005には便利な機能が追加されているため、開発効率を向上させることができます。「PGO (Profile-Guided Optimization)」と呼ばれる最適化機

能も搭載されており、アプリケーション実行時のプロファイルを使用した最適化も可能になっています。

では、これらの新機能のうち、明日からでも使える機能をピンポイントで紹介してゆくことにしましょう。

## コンパイラの拡張

Visual C++ 2005では、より.NET Frameworkとの親和性を高めるために、「C++/CLI」と呼ばれる拡張仕様を採用しています。Visual Studio .NET 2003では、「C++マネージ拡張」という機能を使用して.NET Frameworkを使用したアプリケーションの構築が可能でしたが、新たな機能であるC++/CLIを使用すれば、C#やVisual Basicなどと同様に、完全に.NET Frameworkに対応したアプリケーションを作成することができます。

## C++/CLI

C++/CLIは、マネージコードに完全に対応したアプリケーションを作成するために拡張された言語仕様です。C++マネージ拡張では「\_gc」などのいく

### Level

1 2 3 4 5

### Technology Tools

- Visual Basic
- Visual C#
- Visual C++
- SQL Server
- Oracle
- Access
- Excel
- ASP.NET
- Other:

↓  
Visual Studio 2005

つかのキーワードを追加し、そのキーワードによって.NET Frameworkの機能を使用していました。C++/CLIでは言語仕様そのものを拡張して、.NET Frameworkに対応させています。

リスト1はVisual C++ .NET 2003で作成したWindowsフォームのアプリケーションです。このアプリケーションはC++マネージ拡張を使用しています。C++を知っている人であれば違和感を感じることがないでしょう。

リスト2はVisual C++ 2005で作成したWindowsフォームのアプリケーションです。main関数の引数を見ると、「^」という見慣れない演算子があります。この「^」は、C++/CLIで拡張された部分です。この演算子はマネージヒープ上に配置されたオブジェクトへのハンドルを示し、C++マネージ拡張で使用していた「\_\_gc \*」と同じ意味を持ちます。

それ以外にも、Visual C++ .NET 2003とVisual C++ 2005で作成したWindows

フォームアプリケーションの違いとして、Visual C++ 2005ではスレッドの設定の部分を、

[STAThreadAttribute]

という属性によって設定するように変更されていますし、メインウィンドウを作成する部分では、「new」の代わりに「gcnew」キーワードを使用して、マネージタイプのインスタンスを、ガベージコレクションを行なっているヒープの中に作成しています。

### マネージコード対応アプリケーションのモード

マネージコードに対応したアプリケーションを作成する場合、どの部分をマネージコードに対応させるかによって、

- ・ Mixedモード
- ・ Pureモード
- ・ Verifiableモード

の3つのモードが用意されています。

### Mixedモード

このモードは従来のC++マネージ拡張と同等のモードです。このモードではマネージコードとアンマネージコードを混在させることができます。Mixedモードで作成したアプリケーションにはアンマネージコードが必ず含まれますが、アプリケーションから.NET Frameworkの機能を使用することができます。データも共通言語ランタイムで管理されるマネージデータと、従来のWin 32ベースのアンマネージデータの両方を扱うことができます。しかし、このMixedモードで作成したアプリケーションはVisual BasicやVisual C#で作成したアプリケーションとは違い、あくまでもアンマネージコードで作成されたアプリケーションが基本にあり、その上で.NET Frameworkの機能を使用するという構図になっています。

リスト1：  
Visual C++ .NET 2003で作成したWindowsフォームアプリケーション

```
#include "stdafx.h"
#include "Form1.h"
#include <windows.h>

using namespace VC03_WinForm01;

int APIENTRY _tWinMain(HINSTANCE hInstance,
                      HINSTANCE hPrevInstance,
                      LPTSTR lpCmdLine,
                      int nCmdShow)
{
    System::Threading::Thread::CurrentThread->ApartmentState =
        System::Threading::ApartmentState::STA;
    Application::Run(new Form1());
    return 0;
}
```

リスト2：  
Visual C++ 2005で作成したWindowsフォームアプリケーション

```
// VC_WinForm02.cpp: メインプロジェクトファイルです。

#include "stdafx.h"
#include "Form1.h"

using namespace VC05_WinForm01;

[STAThreadAttribute]
int main(array<System::String ^> ^args)
{
    // コントロールが作成される前に
    // Windows XP ビジュアル効果を有効にします
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);

    // メインウィンドウを作成して、実行します
    Application::Run(gcnew Form1());
    return 0;
}
```