

快速脱! 休日出勤

.NETアプリケーション障害解析の極意

デバッグ技法

第3回

EXE/DLL間のデバッグ —プログラミング工程—

株式会社NTTデータ
飯山 教史
IIYAMA, Takashi

外部モジュールのデバッグ

前回まで開発者自身が開発したコードのデバッグを対象としてきた。しかし、システム開発の現場ではひとりの開発者がすべてを開発するというのは稀であり、第三者が開発したモジュールと連携してシステムを開発するのが普通である。そして、この連携の段階で外部のモジュールのデバッグができずに苦しんでいる開発者をよく見かける。MicrosoftがVisual Studioでモジュール間のデバッグを支援する機能を数

多く提供しているにもかかわらず、これらに気づいていない開発者が多いのはとても残念である。

そこで今回は、EXE/DLL間のデバッグ手法を、「外部モジュールのソースコードを開く」ということに焦点を当てて説明する。これを解説のポイントにしたのは、

Visual Studioで外部モジュールのソースコードを開くことができれば、ブレークポイントを設定してこれまでどおりにデバッグできる

からである。これにより、外部モジュールであることを気にせずにデバッグ作業に集中できるようになる。

EXEとDLLの関係から、外部モジュールとの連携パターンには以下の2通りがある。

- EXEを開発し、DLLのコードを参照する
- DLLを開発し、EXEのコードを参照する

それぞれのデバッグを順番に説明する。

EXEから DLLの内部へ移動する

はじめにEXEからDLLのソースコードを表示する方法を説明する。開発中にDLLを利用するときは、開発が終了したDLLのバイナリを提供してもらい、アプリケーションからそのDLLのメソッドを呼び出すコードを記述する。今回の説明で利用するEXEとDLLのコードはリスト1のとおり単純なものである。

はじめに、SelectModel.dllのバイナリを別のフォルダにコピーしておき、EXEのプロジェクトの「参照の追加」でこのフォルダにあるDLLを指定した状態でデバッグを始める。今回は「binary」フォルダを作成し、そこにDLLをコピーした後に参照設定を行なった(図1)。

もしDLLのロジックが仕様どおりではなく、ReturnModelNameメソッド

レベル >>> Level

1

2

3

4

5

言語 >>> Language

▪ Visual Basic

ツール >>> Tool

▪ Visual Studio 2005 Team Edition
for Software Testers

リスト1：DLLのメソッドを呼び出すEXE

```

'-----
'EXEのコード
'-----

Imports SelectModel

Public Class TopModel
    Private Sub btnDisplayModel_Click( _
        ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles btnDisplayModel.Click
        Dim Name As String = ""
        Dim smn As New SelectModelName()

        ' DLLのメソッドを呼び出す
        smn.ReturnModelName(Name)
        txtModelName.Text = Name
    End Sub
End Class

```

```

'-----
'DLL (SelectModel.dll) のコード
'-----

Public Class SelectModelName
    ' 名前を返すメソッド
    Public Function ReturnModelName(ByRef ModelName As String)
        ' Nullが渡されたらメッセージを表示する
        If ModelName Is Nothing Then
            Return "不正な文字列です。"
        End If

        ' モデル名を返す
        ModelName = "シンディ・クロフォード"
        Return ModelName
    End Function
End Class

```

図1：EXEプロジェクトでSelectModel.dllの参照設定を行なう

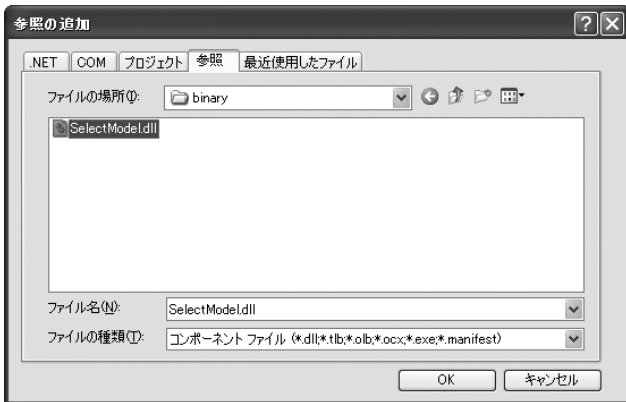


図3：参照設定をDLLプロジェクトのDebugフォルダに変更

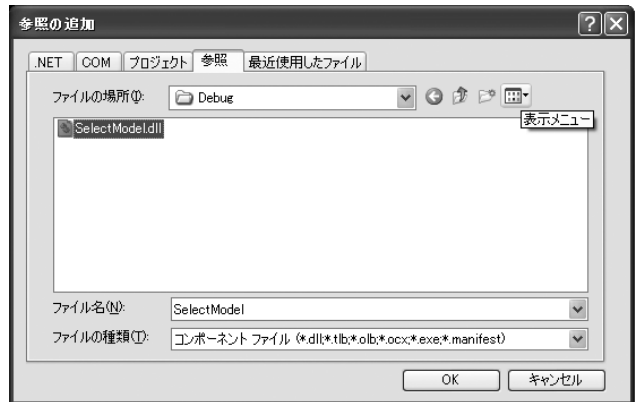
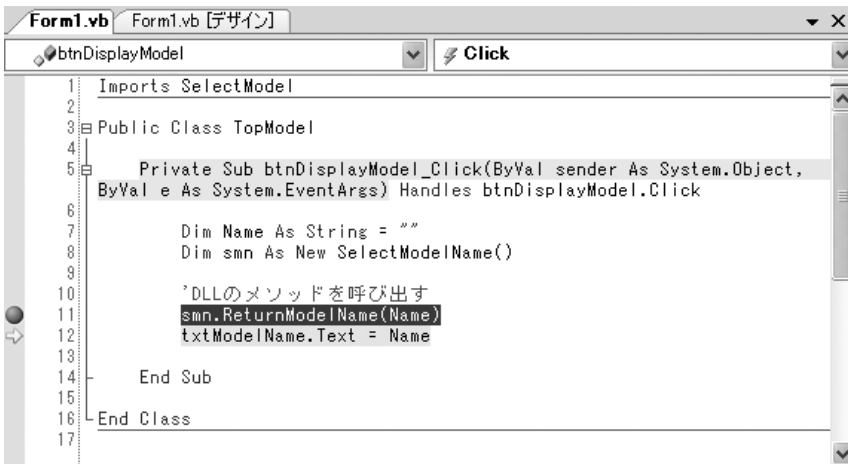


図2：[F11] キーを押しても次行に移動するだけ



から戻される値が予想外のものだった場合はDLLの内部のコードが読みたくなるだろう。だが、ReturnModelNameメソッドの部分で [F11] キーを押してステップインを試みても、EXEのコードで次の行へ移動するだけで、DLLの内部コードは表示されない (図2)。

続いて、DLLを作成したプロジェクトのDebugフォルダに参照を再設定^[注1]する (図3)。DLLのメソッド部分で [F11] キーを押すと「ソースファイルの検索」ウィンドウ (図4) が表示される。ここで、DLLのソースファイルがないので

注1) 再設定する前に、前回の参照を削除しなくてはならない。削除するには、ソリューションエクスプローラの「参照設定」を展開し、削除したい参照 (ここではSelectModel) を右クリックして [削除] を選択する。