

USB カメラプロバイダ

USB カメラ

Version 1.0.0

ユーザーズ ガイド

July 17, 2012

【備考】

本ソフトウェアの著作権はいかなる場合も株式会社セックが保有しています。本ソフトウェアの利用は個人PCのみに限定し、著作者である株式会社セックの許可を得ることなしに、転載、再配付、商業利用などの行為を禁止します。

また、このソフトウェアを使用したことによって生じたすべての障害・損害・不具合等に関しては、株式会社セックおよび、その関係者・団体・組織とも、一切の責任を負いません。本ソフトウェアを利用する個人の責任においてご使用ください。

【改版履歴】

バージョン	日付	内容
1.0.0.0	2006-02-24	初版.
1.0.0.1	2010-02-12	エラーコード追加
1.0.0	2012-07-17	ドキュメントのバージョンルールを変更

【対応機器】

機種	バージョン	注意事項

目次

1. はじめに	4
2. プロバイダの概要	5
2.1. 概要	5
2.2. メソッド・プロパティ	6
2.2.1. CaoWorkspace::AddController メソッド	6
2.2.2. CaoController::OnMessage イベント	6
2.2.3. CaoController::AddVariable メソッド	7
2.2.4. CaoController::AddFile メソッド	7
2.2.5. CaoVariable::get_Value	7
2.2.6. CaoVariable::put_Value	7
2.2.7. CaoFile::get_Value	7
2.3. エラーコード	8
3. サンプルプログラム	9

1. はじめに

本書は、ORiN アプリケーションで、USB カメラを利用するための CAO プロバイダである、USB カメラプロバイダのためのユーザーズガイドです。

USB カメラプロバイダでは、Microsoft 社の Windows 用マルチメディア拡張 API 群 DirectX に含まれる API の 1 つである DirectShow を利用し、USB カメラから画像を取得しています。

USB カメラプロバイダには以下の機能を持っています。

- 一定周期で USB カメラから画像を取得する。
- クライアントからの要求に従い、USB カメラの画像をクライアントに渡す。
- カメラ画像に変化が生じた場合、それを検知し、クライアントに通知する変化を検知する際の感度を設定する。

この USB カメラプロバイダを利用することにより、ハードウェアの違いを気にすることなく、画像の変化を検出して、ユーザにメール通知するようなアプリケーションが容易に構築できるようになります。さらに、DCOM や CAP プロバイダを利用することにより、リモート PC 上の USB カメラにアクセスすることができるので、遠隔監視システムも容易に開発することができます。

2. プロバイダの概要

2.1. 概要

USB カメラプロバイダは、OnTimer イベントを利用し、一定周期(10msec)で、USB カメラから画像を取得し、画像に変化が発生するとイベントを通知します。また、ファイルクラスを利用し、USB カメラ画像の取得をおこなうことができます。さらに、変数クラスを利用し、画像変化イベントを通知する際の反応感度を取得/設定することができます。

USB カメラプロバイダのファイル形式は DLL(Dynamic Link Library)となっており、その詳細は表 1 のようになっています。

表 1 USB カメラプロバイダ

ファイル名	Camera.dll
ProgID	GaoProv.Samples.Camera
レジストリ登録	regsvr32 Camera.dll
レジストリ登録の抹消	regsvr32 /u Camera.dll

また、本プロバイダが対象としている USB カメラは、Windows Driver Model(WDM)というドライバ規格に対応している必要があります。お持ちの USB カメラが WDM 対応であることをご確認ください。なお、USB カメラから画像を取得するためには、DirectX SDK が必要となります。事前に Microsoft 社のサイト(<http://www.microsoft.com/downloads/>)からダウンロードし、インストールをおこなってください。

2.2. メソッド・プロパティ

2.2.1. CaoWorkspace::AddController メソッド

USB カメラプロバイダでは、AddController で USB カメラへの接続をおこないます。

ここで、USB カメラが見つからない場合や、すでに USB カメラに接続済みの場合は、エラーとなります。また、PC に複数の USB カメラを接続している場合は、USB カメラプロバイダが最初に見つけた USB カメラに接続をおこないます。

以下に、AddController メソッドの引数仕様を示します。

```
AddController
(
    "<Controller 名>",           // コントローラ名
    "CaoProv. Samples. Camera", // プロバイダ名. 固定.
    "<マシン名>",             // プロバイダの実行マシン名.
    "<オプション>"            // オプション文字列
)
```

ここで、Controller 名には、任意の文字列を指定してください。オプション文字列には、何も指定する必要はありません。以下に AddController メソッドを実行例を示します。

```
AddController
(
    "Cam1",                       // コントローラ名 = Cam1
    "CaoProv. Samples. Camera",   // CAO エンジンプロセスで実行
    "",
    ""
);
```

2.2.2. CaoController::OnMessage イベント

USB カメラプロバイダでは、USB カメラから取得した最新の画像と、1 サンプル前に取得した画像を常に比較しています。この 2 枚の画像の変化量がある閾値を超えた場合、OnMessage イベントを発行します。

このイベントでは、以下のプロパティを持つメッセージを受け取ることができます。

DateTime プロパティ : 画像変化イベントが発生した時間
Value プロパティ : メッセージ文字列

2.2.3. CaoController::AddVariable メソッド

USB カメラプロバイダでは、CaoVariable クラスを利用して、画像変化検出の閾値を設定/取得することができます。

以下に、AddVariable の引数仕様を示します。

```
AddVariable  
(  
    "<変数名>"           // 変数名  
    ""                   // オプション文字列 (空文字列)  
)
```

ここで、変数名には、任意の文字列を指定してください。オプション文字列は使用しません。

2.2.4. CaoController::AddFile メソッド

USB カメラプロバイダでは、CaoFile クラスを利用して、USB カメラの画像を取得することができます。

以下に、AddFile の引数仕様を示します。

```
AddFile  
(  
    "<ファイル名>"       // ファイル名  
    ""                   // オプション文字列 (空文字列)  
)
```

ここで、ファイル名には、任意の文字列を指定してください。オプション文字列は使用しません。

また、ここで取得した CaoFile は、読み取り専用であり、値を設定することができないので注意してください。

2.2.5. CaoVariable::get_Value

CaoVariable クラスを利用して、画像変化イベントの閾値を取得することができます。取得した出たは、VARIANT の VT_I4 型で取得することができます。

2.2.6. CaoVariable::put_Value

CaoVariable クラスを利用して、画像変化イベントの閾値を設定することができます。設定する値には、4 バイト整数型を指定してください。

2.2.7. CaoFile::get_Value

CaoFile クラスを利用して、USB カメラ画像を取得することができます。画像データは、VARIANT の VT_UI1 の配列型として取得することができます。ここで取得したデータは、ビットマップファイル形式のバイナリデータとなります。

2.3. エラーコード

USB カメラプロバイダでは、固有のエラーコードはありません。ORiN2 共通エラーについては、[「ORiN2 プログラミングガイド」](#)のエラーコードの章を参照してください。

3. サンプルプログラム

以下に、ボタンを押下することにより、ピクチャボックスに USB カメラから取得した画像が表示されるサンプルプログラムを示します。

なお、本サンプルプログラムは、Visual Basic 2005 にて作成しました。

List 3-1 Form1.vb

```
Imports System.io

Public Class Form1
    Inherits System.Windows.Forms.Form
    ' CAO エンジン
    Private caoEng As CaoEngine
    ' CAO ワークスペース
    Private caoWS As CaoWorkspace
    ' USB カメラの CAO コントローラ
    Private ctrlCamera As CaoController
    ' カメラ画像の CAO ファイル
    Private m_filePicture As CaoFile

    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        ' CAO エンジンと CAO ワークスペースの作成
        caoEng = New CaoEngine
        caoWS = caoEng.Workspaces.Item(0)
        ' USB カメラプロバイダへの接続
        ctrlCamera = caoWS.AddController("Cam1", "CaoProv.Samples.Camera", "", "")
        ' CAO ファイルの取得
        m_filePicture = ctrlCamera.AddFile("Picture")
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles Button1.Click
        Dim bArray() As Byte
        ' 画像データの取得
        bArray = m_filePicture.Value
        ' バイナリデータから MemoryStream オブジェクトを作成する
        Dim mstreamCur As New MemoryStream(bArray)
        ' ストリームから Image オブジェクトを作成し、ピクチャボックスに表示
        PictureBox1.Image = Image.FromStream(mstreamCur)
    End Sub
End Class
```

また、サンプルアプリケーションの実行例を図 1 に示します。



図 1 サンプルプログラム実行例