

シリアル番号検証を行なうダイアログの作成

検証したバージョン: InstallShield 2009 Premier Edition

対象プロジェクト: 基本の MSI プロジェクト

概要

インストーラのダイアログより、シリアル番号検証を行う場合、検証を行う外部 DLL ファイルの作成、ダイアログやコントロールに対する設定やカスタムアクションの作成が必要となります。

この記事では、「CustomerInformation」ダイアログの MaskedEdit フィールド「SerialNumber」を利用してシリアル番号の検証を行うインストーラの設定手順について簡単に説明します。

A. シリアル番号の検証を行う DLL ファイルの作成

シリアル番号の検証のロジックは、セキュリティの観点上インストーラ開発者が独自の仕組みとして作成する必要があります。一般的には外部 DLL ファイルの形式で作成したものを、インストーラよりカスタムアクションとして実行する方法等が用いられます。

シリアル検証を行う DLL ファイルのプロジェクトサンプルは、以下の箇所より入手可能です。

<http://cabi.networld.co.jp/faq/installshield/tips/ValidateSerialNumber.zip>

また、InstallShield のバージョンによっては、以下の階層に既定でサンプルプロジェクトが配置されています。

`<ISProductFolder>¥Samples¥WindowsInstaller¥ValidateSerialNumber`

上記のサンプルに加えて、DLL ファイルより msi 形式ファイルが使用するプロパティの取得を行なう場合、Windows Installer より提供されている API 「MsiGetProperty」および「MsiSetProperty」が使用可能です。これらの API の詳細は製品ヘルプ「プロパティの取得と設定」をご参照ください。

この記事では、以下の関数を持つ シリアル番号検証用の Win32 DLL ファイルを使用すると仮定します。

関数名: *ValidateSN*
引数: 検証するシリアル番号 (文字列)
戻り値: 検証が成功した場合「1」 検証が失敗した場合「-1」

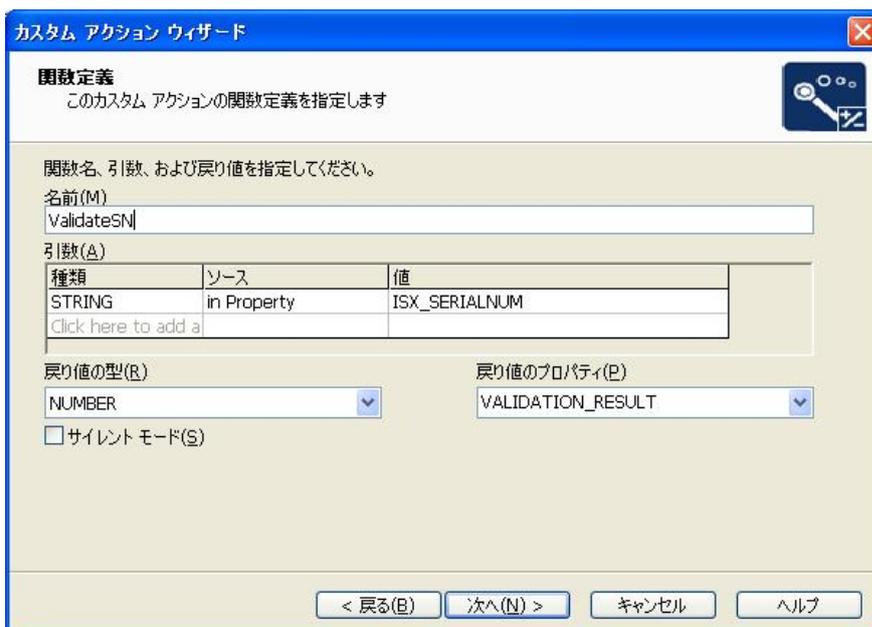
B. シリアル番号検証を行うカスタムアクションの作成

InstallShield 上でカスタムアクションの設定を行ないます。IntallShield にて新規に「基本の MSI」プロジェクトを作成して、[動作とロジック]-[カスタム アクションとシーケンス]ビューを選び、カスタムアクションのアイコンを右クリックして、[カスタム アクション ウィザード] を選択します。カスタム アクション ウィザードが起動しますので、以下のとおりに入力を行ないます。

1. 基本情報パネルにて、[名前]にカスタムアクションの名称として「caValidateSerial」と入力します
2. アクションの種類 パネルにて[種類]に[標準ダイナミックリンク ライブラリの関数を呼び出す]を指定します。[場所]に[Binary テーブルに保存する]を指定します。
3. 関数定義 パネルにて、以下の設定を行ないます。

入力箇所	設定する文字	説明
名前	ValidateSN	DLL ファイル内で定義した関数名を指定します。
引数	種類:STRING ソース:in Property 値:ISX_SERIALNUM	関数への引数となるプロパティを指定します。 ISX_SERIALNUM は、CustomerInformation ダイアログにデフォルトで配置されている Masked edit の入力値が格納されるプロパティです。
戻り値の型	NUMBER	戻り値の型を指定します。
戻り値のプロパティ	VALIDATION_RESULT	関数からの返却値を保持するプロパティを指定します。 このプロパティには任意の名称が指定可能です。

4. アクションのパラメータパネルでは、[ソース]に作成した DLL ファイルの階層を指定します
5. それ以外のパネルは、既定の設定を使用します



C. シリアル検証失敗時にエラー表示を行なうカスタムアクションの作成

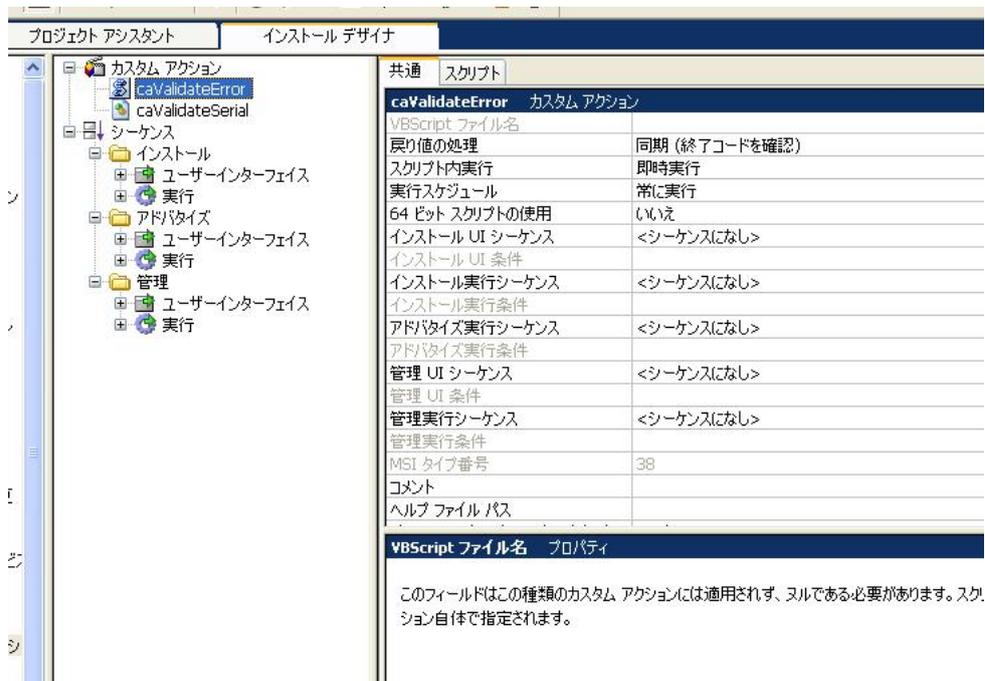
DLL 内の関数でシリアル番号の判定を行った後、シリアルが不正だった場合にユーザへエラー表示を行なうカスタムアクションを作成します。エラー表示の仕組みはシリアル番号の検証を行う DLL 自体に組み込んでいただいても、もちろん問題ありませんがこの記事では VBScript カスタムアクションを使用して行ないます。

[カスタム アクションとシーケンス]ビューにて、カスタムアクションのアイコンを右クリックして、再度 [カスタム アクション ウィザード] を選択します。カスタム アクション ウィザードが起動しますので、以下のとおりに入力を行ないます。

1. 基本情報パネルにて、[名前]にカスタムアクションの名称として「caValidateError」と入力します
2. アクションの種類 パネルにて[種類]に[VBScript コードを実行する]を指定します。[場所]に [カスタム アクションに直接保存する]を指定します。
3. シーケンス内スクリプトパネルにて、実行を行なう VBScript を直接記述します。以下はエラー表示を行なう VBScript の一例です。

MsgBox "シリアル番号が間違っています",vbOKOnly,"シリアル検証エラー"

4. それ以外のパネルは、既定の設定を使用します



D. ダイアログの設定

ダイアログにシリアル番号の入力を行なうコントロールを追加して、[次へ]ボタンの押下時にシリアル番号を検証するカスタムアクションを呼び出す設定を行ないます。

[ユーザーインターフェイス]-[ダイアログ]ビューにて、右に表示されるダイアログの一覧より [CustomerInformation]を選択し、以下の設定を行ないます。

1. [CustomerInformation]-[日本語]を選び、右ウィンドウのダイアログエディタにて、既存の MaskedEdit コントロール[SerialNumber]を選択して、以下の設定を行ないます。

Mask: ?????-####-#### 等

Visible: True

※Mask プロパティには、シリアル番号の入力形式を指定します。“?”は半角英数字、“#”は半角数字のみの入力を許可します。上記例の場合 ABC123-4567-8910 のような入力が可能です。

Mask プロパティに指定する文字の詳細は、製品ヘルプ「MaskedEdit Control」をご参照ください。

2. [CustomerInformation]-[動作]を選び、コントロールの一覧より[Next]を選択します。

3. 右ウィンドウ下部のタブを[イベント]に切り替え、デフォルトで存在するイベントの内下段二行を削除して、以下の三つのイベントを追加します。

イベント	引数	条件	説明
DoAction	caValidateSerial	1	シリアル番号検証を行うカスタムアクション caValidateSerial を実行します。
NewDialog	SetupType	VALIDATION_RESULT > 0	caValidateSerial の返却値が正常だった場合、次のダイアログを表示します。
DoAction	caValidateError	VALIDATION_RESULT < 0	caValidateSerial の返却値が異常だった場合、エラー表示のカスタムアクション caValidateError を実行します。

イベント	引数	条件
[ALLUSERS]	1	ApplicationUsers = "AllUsers" And Privileged
[ALLUSERS]	{}	ApplicationUsers = "OnlyCurrentUser" And Priv
DoAction	caValidateSerial	1
NewDialog	SetupType	VALIDATION_RESULT > 0
DoAction	caValidateError	VALIDATION_RESULT < 0
Click here to add a new item		

E. ビルドおよび動作の確認

メニューバー[ビルド]ー[ビルド(B)]または、F7 ボタンよりプロジェクトのビルドを行ないます。

作成されたインストーラを実行すると、「ユーザー情報」ダイアログにシリアル番号の入力を行なうフィールドが追加されており、「次へ」ボタンの押下時カスタムアクションが実行されてシリアル番号の検証が行われます。

シリアル番号の検証に失敗した場合、エラーが表示されます



シリアル番号の検証に成功した場合、次のダイアログが表示されます

