

Windows Installer 形式 DLL ファイルの作成

この文書は Acrezzo Software の次の文書を元に記載しています。

<http://www.acresso.com/webdocuments/PDF/dlls-for-ipwi.pdf>

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

概要

InstallShield 2009 のカスタムアクションウィザードでは、「Windows Installer のダイナミック リンクライブラリの関数を呼び出す」機能がサポートされています。この記事ではこのタイプのカスタムアクションを作成する際に、関連する手順と注意点について説明します。

カスタムアクションより呼び出しを行なう DLL ファイルの作成

Windows Installer 形式の DLL は以下の宣言により、関数のエクスポートを行ないます:

```
UINT __stdcall ActionName(MSIHANDLE);
```

以下に、Microsoft Visual C++ のサンプルコードを記載いたします。以下の二つのファイル「CustomAction.cpp」「CustomAction.def」を Microsoft Visual Studio 等を使用して作成し、ビルドを行い、Windows Installer 形式の DLL ファイル「CustomAction.dll」を作成してください。

ファイル: 「CustomAction.cpp」

```
// link to MSI library
```

```
#pragma comment(lib, "msi.lib")
```

```
// include standard Windows and MSI headers
```

```
#include < windows.h >
```

```
#include < msi.h >
```

```
#include < msiquery.h >
```

```
// code for first action; note that the function name FirstAction
```

```
// is the name we enter in the Custom Action Wizard
```

```
UINT __stdcall FirstAction(MSIHANDLE hInstall)
```

```
{
```

```
    MessageBox(  
        GetForegroundWindow( ),  
        TEXT("This is FirstAction."),  
        TEXT("Custom Action"),  
        MB_OK | MB_ICONINFORMATION);
```

```
    return ERROR_SUCCESS;
```

```
}
```

ファイル:「CustomAction.def」

```
LIBRARY CustomAction
EXPORTS
    FirstAction
```

カスタムアクションの作成

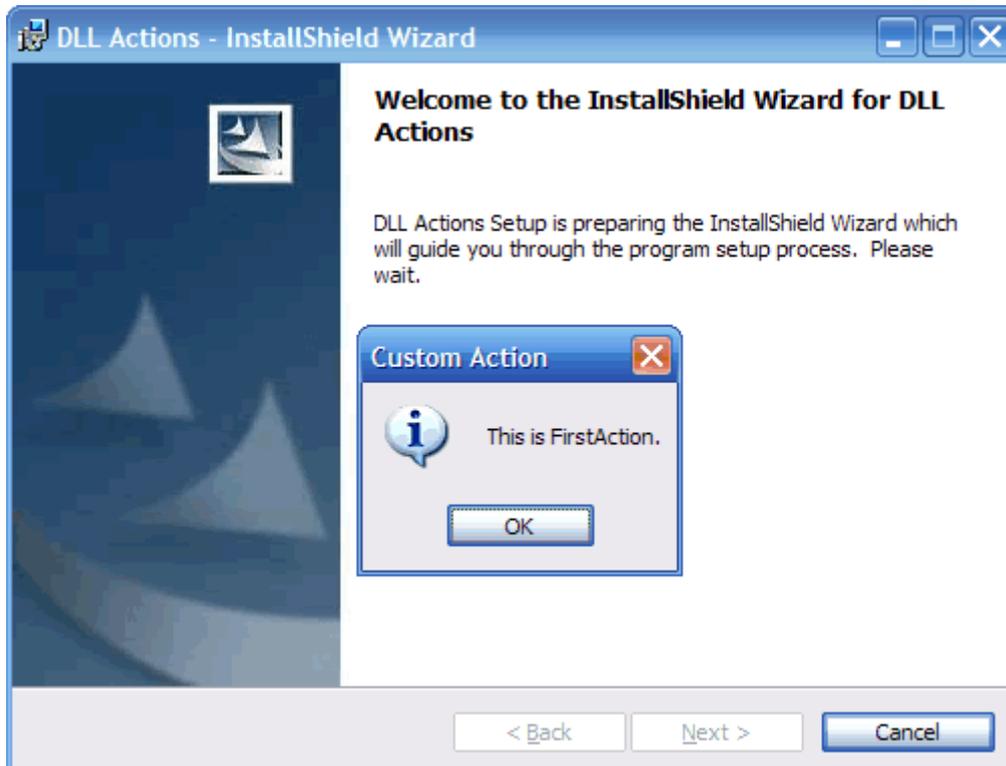
次に InstallShield 上でカスタムアクションの設定を行ないます。InstallShield にてプロジェクトを開いて [インストール デザイナ] タブ - [動作とロジック] - [カスタム アクションとシーケンス] ビューを選び、カスタムアクションのアイコンを右クリックして、[カスタム アクション ウィザード] を選択します。カスタム アクション ウィザードが起動しますので、以下の通りに入力を行ないます。

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「FirstAction」と入力します
2. アクションの種類パネルでは、[種類]に「Windows Installer のダイナミック リンク ライブラリの関数を呼び出す」を選び、[場所]は「Binary テーブルに保存する」を選択します
3. 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て、[ターゲット]には、作成した関数名「FirstAction」を指定します
4. 追加パネルは、既定の設定を使用します

最後に インストーラの User Interface シーケンスにカスタムアクションを挿入します。

[インストールデザイナ] - [動作とロジック] - [カスタムアクションとシーケンス] ビューを選択して [シーケンス] - [インストール] - [ユーザーインターフェイス] を展開します。 [AppSearch] を右クリックして、[挿入]を選び、[FirstAction]カスタムアクションを選択します。

プロジェクトをビルドして実行させると、DLL からのメッセージが表示されます。



インストーラ と DLL カスタムアクション間の情報の受け渡し

Windows Installer DLL 形式の関数は、MSIHANDLE 型の一つの引数しか許可しません。
(これには、実行中のインストーラのハンドルが含まれます)

DLL に追加の情報を引き渡すための、追加の引数は使用できません。
(しかしながら、「標準ダイナミックリンクライブラリの関数を呼び出す」タイプのカスタムアクションでは別の形式の関数を呼び出すことを許可するため、任意の引数を指定することが可能です。)

実行中のインストーラと DLL 関数の間で、情報の受け渡しを行なう場合は、Windows Installer API の MsiGetProperty または MsiSetProperty を使用して、Windows Installer のプロパティを取得・設定を行なうことで対応します。

二つ目に作成するカスタムアクションは、デフォルトで定義されてるプロパティ[USERNAME] のデフォルト値の読み込み、表示、変更 を行なうサンプルとなります。

以下に二つ目のカスタムアクションのサンプルのコードを記載いたします。先ほど作成した DLL ファイルのプロジェクトに以下のコードを追加してください。

ファイル:「CustomAction.cpp」

```

UINT __stdcall SecondAction(MSIHANDLE hInstall)
{
    // max. USERNAME size, from CustomerInformation panel

    const DWORD MAX_USERNAME_LEN = 50;
    // initialize empty string and size buffer

    TCHAR UserName[MAX_USERNAME_LEN + 1] = {0};
    DWORD Buffer = MAX_USERNAME_LEN + 1;
    // read and display the initial value of USERNAME

    MsiGetProperty(hInstall, TEXT("USERNAME"), UserName, &Buffer);
    MessageBox(
        GetForegroundWindow( ),
        UserName, TEXT("USERNAME"),
        MB_OK | MB_ICONINFORMATION);

    // change the value of USERNAME
    MsiSetProperty(hInstall, TEXT("USERNAME"),
        TEXT("Valued Customer"));

    return ERROR_SUCCESS;
}

```

ファイル:「CustomAction.def」

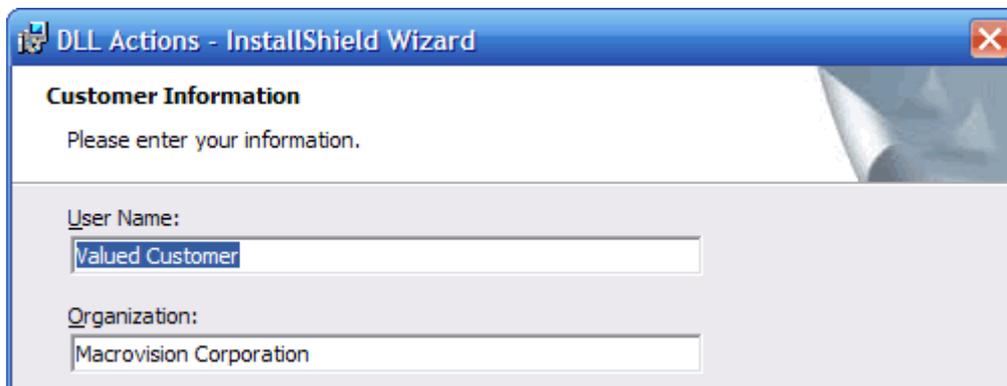
```

LIBRARY CustomAction
EXPORTS
    FirstAction
    SecondAction

```

前回と同様に DLL をビルドします。「SecondAction」という名称のカスタムアクションをカスタムアクションウィザードを使用し、作成して「FirstAction」カスタムアクションの直後に挿入します。

インストーラを再度ビルドして実行すると、二つ目のメッセージボックスに デフォルトの USERNAME プロパティが表示されて、[ユーザ 情報]ダイアログでは、ユーザ名に表示されるプロパティの値が変更されています。



プロパティを使う上の問題

Windows Installer の プロパティを使用する場合は、以下の内容に注意してください。

- ・ Msi SetProperty は、実行時にプロパティを新規で作成することも可能です。
- ・ ユーザーインターフェイスシーケンスでプロパティをセットして、実行シーケンスでそれを読み込む場合は、必ずパブリックプロパティを使用してください。パブリックプロパティとは、名称に大文字だけが使用されているプロパティです。(パブリックプロパティの例: USERNAME)

プライベートプロパティとは、名称に小文字が使われており、インストールの処理が ユーザーインターフェイスから、実行シーケンスに変わった時にデフォルト値がリセットされます。

(プライベートプロパティの例: Installed)

- ・ プロパティの取得・変更を行なっているカスタムアクションは、“即時実行”のカスタムアクションとしてスケジュールしてください。遅延実行の際にプロパティの値を取得する方法については、以下の記事をご参照ください。

Deferred Execution Custom Action Cannot Retrieve a Property Value

<http://support.installshield.com/kb/view.asp?articleid=Q104413>