

Windows Vista時代の デバイス・ドライバ開発

日高 亜友,
川出 智幸,
相良 徹

第16回/最終回 DIFx ツールを使用したインストーラ作成

Windows デバイス・ドライバのインストーラ作成ツール「DIFx ツール」を使用し、ドライバのインストール関連操作を記述するためのC言語ライブラリであるDIFxAPIを利用したインストール用アプリケーションの作成方法について解説する。

本連載では、Windows Vistaで導入されたドライバ開発手法に関して取り上げてきたが、新世代OS「Windows 7」が発表され、Windows 7用WDK(ベータ版)の配布も開始された。これを機に、本連載は最終回とする。(筆者)

前回(2009年1月号, pp.172-177)に引き続き、Windows デバイス・ドライバのインストーラ作成ツールであるDIFx ツール「DIFxAPI (DIFxLib)」について解説します。次世代OSであるWindows 7が発表されましたが、WDKとそれに含まれるDIFx ツールは、ドライバのインストーラを開発するために引き続き使用されていきます(コラム1)。

1. C言語ライブラリ——DIFxAPI

● DIFxAPI とは

DIFxAPIは、VC++を使用してドライバのインストール関連の操作を記述するためのC言語ライブラリです。コンパイラはVisual Studioではなく、WDKとともに配布されるものを使用します。ドライバのインストール関連の操作を記述するライブラリ群は、図1のように分類できます。

広義のDIF(Device Install Function) APIとしては、General Setup関数、Device Installation関数、PnP Configuration Manager関数、DIFxAPI関数と、各種の構造体、マクロ類の定義やプロトタイプ宣言が含まれます。

① General Setup 関数

Setupのプレフィックスで始まる名前を持ちます。SDKにも含まれるアプリケーション用のインストーラ開発でも利用される、一般的なインストーラ開発用APIを提供します。

② Device Installation 関数

ほとんどがSetupDiプレフィックスで始まる名前です。General Setup関数では扱えない、デバイス・インストール専用の機能を提供します。

③ PnP Configuration Manager 関数

CM_のプレフィックスで名前が始まります。プラグ&プレイ関連の機能を提供します。

④ DIFxAPI 関数

これは、DPInst相当のインストーラを記述するための機能をまとめて、インストーラを記述しやすいAPIとして提供します。各関数やマクロ定義の詳細仕様は、WDKのHelp DocumentやMSDN Library(英語版)に記載されています。

● DIFxAPI 関数を利用したサンプル・コード

WDKのsrc¥setup¥DIFxAPI以下には、DIFxAPI関数を使用したインストール用アプリケーションのサンプル・コードがあります。DIFxCmdはDIFxAPI関数の使用例を示し、AppDrvは各ドライバ固有のINSTALLER INFO構造体の利用方法を示しています。

どちらのサンプルも、プラグ&プレイ型ドライバのインストール操作やドライバ・ストアへのプレインストール操作に対応します。しかしDPInstとは異なり、コントロールパネルの「プログラムと機能」のエントリ(登録情報を表示させ、削除操作を可能にする)は作成しません。

このサンプルのままでは、DPInstよりも機能が劣って

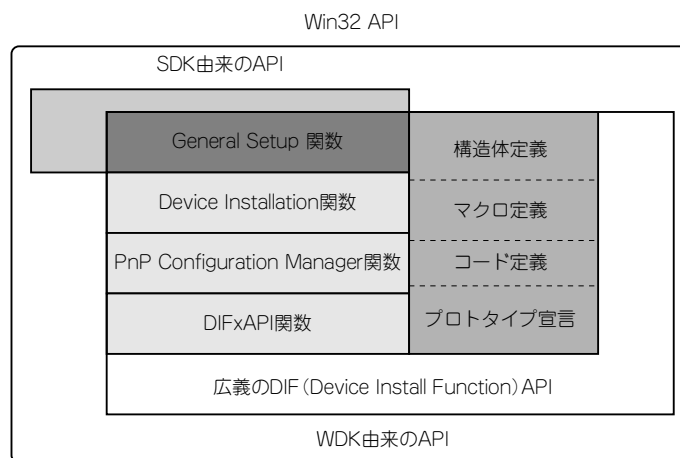


図1 ドライバ・インストーラ開発用APIライブラリの構成図

コラム1 | Windows 7の最新情報

2008年秋にロサンゼルスで開催された、PDC (Professional Developers Conference) 2008と WinHEC (Windows Hardware Engineering Conference) 2008で、Windows 7と Windows Server 2008 R2が正式に発表され、関連情報が公開されました。

筆者らが参加した WinHEC 2008では、次期 OSである Windows 7用のハードウェアやデバイス・ドライバ開発に関する多くの個別セッションが行われました。その中の「Creating Deployable Driver Packages for Windows」のセッションを踏まえて、Windows 7の時代に期待される今後のドライバ・パッケージ(インストーラ)の開発について、三つのポイントを説明します。

● Windows Update での配布

今までデバイス・ドライバは、デバイスとは別に CD などのメディアで提供されていました。利用者に各デバイスのインストール用 CD を保管させるのは利便性が悪く、またドライバや OS のアップデートにより、新しいバージョンのドライバを再配布する必要がありました。

2008年から Windows ログ取得済みのドライバであれば、Windows Update でデバイス・ドライバを配布できるようになりました。これにより、デバイスのインストール用 CD やインストール・マニュアルを製品に添付する必要がなくなります。Windows Update によるネットワーク・インストールが、今後のデバイス・ドライバ配布方法の中心となっていくでしょう。各ベンダは、Windows Update を利用したデバイス・ドライバ配布方法を検討すべきです。

● アプリケーションの配布方法

ベンダが製品を販売する際に、そのデバイス専用のアプリ

ケーションを添付する場合があります。そのような場合は、アプリケーションのインストーラにドライバのインストール機能を含めるのではなく、ドライバのインストーラ(ドライバ・パッケージ)にアプリケーションのインストール機能を含めるべきです。

本連載で紹介するコインストーラの機能を利用すると、プラグ&プレイ・デバイスが使用される時(装着時)に、アプリケーションをインストールすることが可能になります。またこの方法は、デバイス専用アプリケーションを Windows Update での配布用パッケージとして作成できます。

● インストーラを極力シンプルにする

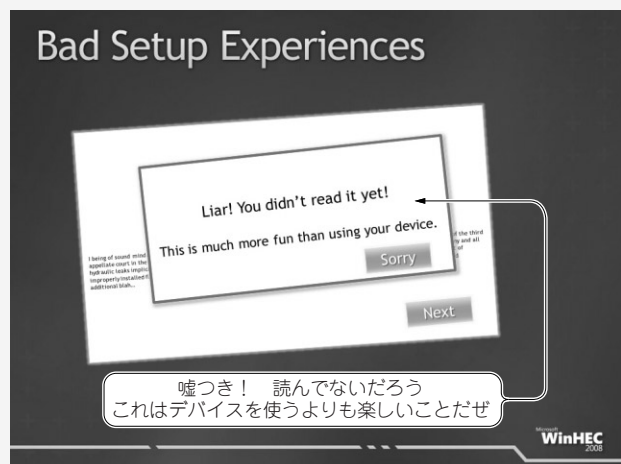
セッション中に、利用者を惑わせたりするさまざまな問題点が指摘されました。8ステップの手順で紹介する「Bad Setup Experiences」の例では、使用ライセンス許諾を表示して読ませ、次に進む場面では「嘘つき！読んでないだろ」と表示して、聴衆を失笑させていました(図A、図B)。

本セッションを行った Eugene Lin 氏は、「結局のところ、ただデバイスを接続するだけで、それが利用できればよい。ドライバがインストールされたことを利用者に気付かせないのが最良のインストーラだ」と言い切っています。これは、Microsoft 社が現在提供している、WDK のインストーラ開発キットである、DIFx ツール群を使用して開発するインストーラの標準的な動作をも否定しているのですが、デバイス・ドライバ本来の裏方としての役割を、再認識させられました。

ほかにも多くの有用な WinHEC 2008 の資料が WinHEC の Web サイト (<http://www.microsoft.com/japan/whdc/winhec/>) で公開されているので、ぜひ参照してください。



図A Bad Setup Experiences (悪いセットアップの事例1)



図B Bad Setup Experiences (悪いセットアップの事例2)