

## <インストール実行時に、Windows の機能を有効化する方法>

注)このドキュメントは、*InstallShield 2014 Premier Edition* を基に作成しています。*InstallShield 2014* 以外のバージョンでは設定名などが異なる場合もあります。

### 概要

インストーラーの実行中に、IIS(インターネット インフォメーションサービス)などの「Windows の機能」を有効化した場合があります。

InstallShield の Premier Edition で提供されるスイート/アドバンスドUI プロジェクトでは、Windows の機能を有効化するための機能があり、製品インストーラーの実行時に Windows の機能を有効化することができます。

一方、Professional Edition では、Windows の機能を有効化するための機能は利用できないため、「展開イメージのサービスと管理 (DISM.exe) ツール」を使用して、コマンドをカスタムアクションから呼び出すことで、インストール時に有効化させることが可能です。

DISM ツールに関する詳細は、以下をご参照ください。

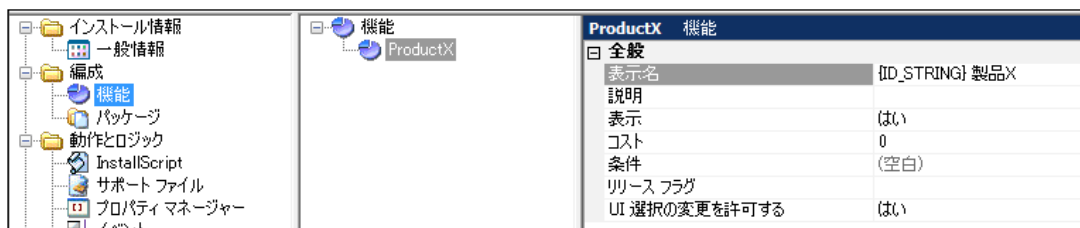
DISM を使って Windows の機能を有効または無効にする  
<https://msdn.microsoft.com/ja-jp/library/hh824822.aspx>

この記事では、それぞれの Edition で「Windows の機能」を有効化する方法を紹介します。

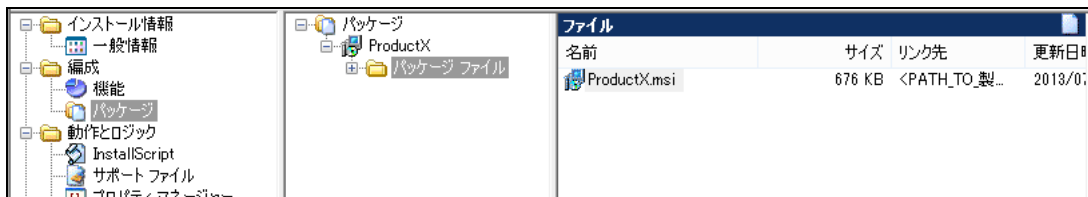
### A. Premier Edition のスイート/アドバンスドUIプロジェクトを使用する場合

スイート/アドバンスドUI プロジェクトを使用すると、製品自体のインストーラーを変更することなく、Windowsの機能の有効化を行うことができます。今回は、IIS(インターネット インフォメーションサービス) を有効化する手順を紹介します。

1. スイート/アドバンスド UI プロジェクトを作成します
2. 機能を作成し、機能の内部名と[表示名]を設定します (内部名:ProductX、表示名:製品 X など)



また、含める.msi パッケージを追加します



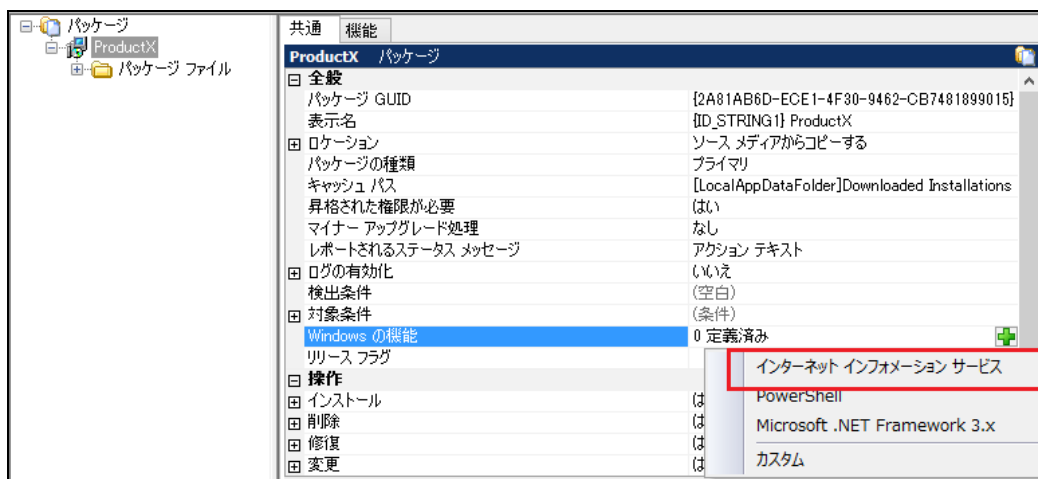
### ●スイートインストーラーの基本的な作成方法

Tips and Tricks (<http://www.networld.co.jp/is/tips.htm>) から以下の記事をご参照ください。

#### [スイートインストーラーの基本的な作成方法](http://www.networld.co.jp/is/pdf/SuiteInstaller_Basic.pdf)

[http://www.networld.co.jp/is/pdf/SuiteInstaller\\_Basic.pdf](http://www.networld.co.jp/is/pdf/SuiteInstaller_Basic.pdf)

- 追加したパッケージ名を選択して、[共通]タブを表示します
- [Windows の機能] の緑の[+]ボタンから、[インターネット インフォメーションサービス]を選択します



有効化したい機能名が追加されたことを確認します

Windows の機能	1 定義済み
Windows の機能	<u>IIS-WebServer Role</u>

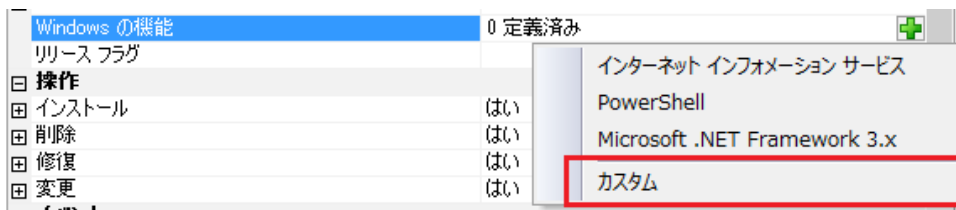
- ビルドして実行し、IIS が有効化されたことを確認します

**●ビルトインで提供されていない Windows 機能の有効化**

スイートアドバンスドUIプロジェクトでは、以下の3つの Windows の機能がビルトインでサポートされています。

- インターネットインフォメーションサービス
- PowerShell
- Microsoft .NET Framework 3.x

これ以外の Windows 機能を有効化するには、[カスタム]を選択したあと、Windows の機能名を直接入力します



例) Telnet クライアントを有効化する場合

Windows の機能	1 定義済み
Windows の機能	TelnetClient

機能名は、以下のような /Get-Features オプションで確認することができます。

Dism /online /Get-Features

## B. Professional Edition でカスタムアクションを使用してDISMコマンドを実行する場合

InstallShield の Professional Edition ではスイート/アドバンスト UI プロジェクトは提供されていないため、方法 A を利用することができません。そのため、Professional Edition では、製品のインストーラー自体にカスタムアクションなどを組み込む必要があります。Premier Edition でも、スイート/アドバンスト UI プロジェクトを使用したくない場合、方法B を利用できます。

Windows の機能は、以下のDISM コマンドを実行することで有効化できます。

Dism /online /enable-feature /FeatureName: <機能名>

ここでは、IIS(インターネット インフォメーションサービス) を有効化するコマンドを .bat ファイルに記述して、そのバッチファイルを実行させることで、Windows の機能を有効化させる方法をご案内します。

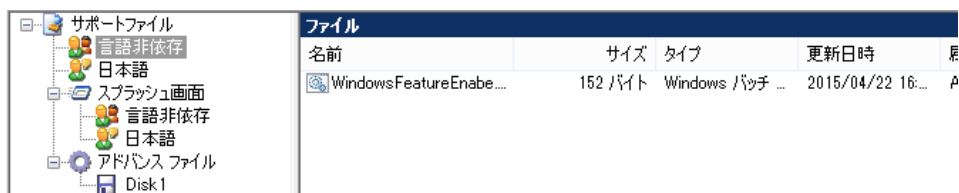
### <基本の MSI/ InstallScript MSI の場合>

1. バッチファイルを作成します

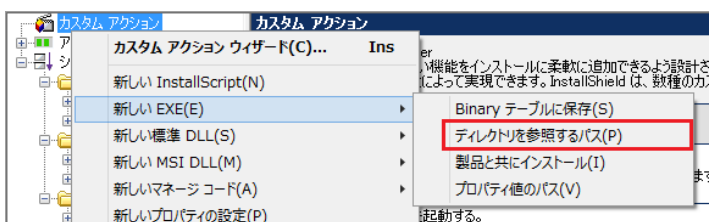
例: WindowsFeatureEnable.bat の内容

```
@echo off
echo IIS を有効化しています。このウィンドウは閉じないでください。
%windir%\system32\dism.exe /online /quiet /enable-feature /featurename:IIS-WebServerRole
```

2. 製品のプロジェクトを開き、[動作とロジック]—[サポート ファイル] ビューにて、[サポートファイル] のツリーを展開して、[言語非依存] を選択します
3. [ファイル]ペインを右クリックして、[ファイルの挿入]を選択します。ファイル選択のダイアログで、“WindowsFeatureEnable.bat”を選択します
4. “WindowsFeatureEnable.bat” がサポートファイルとしてインストーラーに追加されたことを確認します

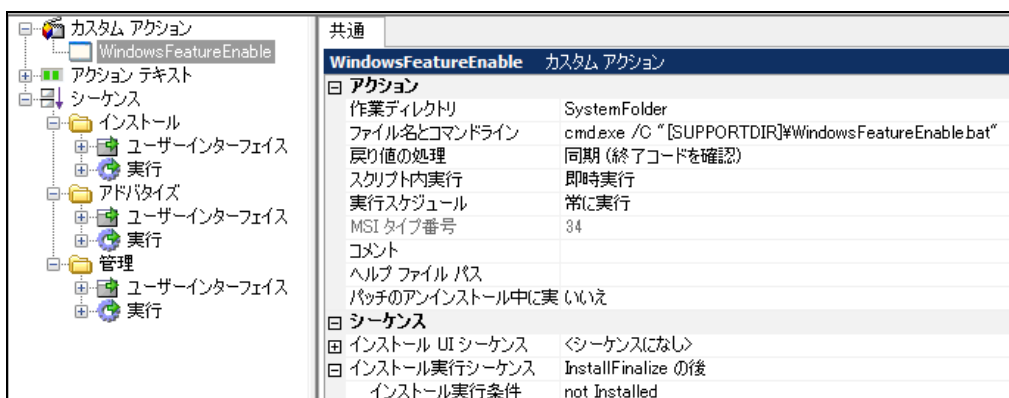


5. [動作とロジック]—[カスタム アクションとシーケンス]ビューを表示し、[カスタムアクション]アイコンを右クリックして、[新しい EXE]—[ディレクトリを参照するパス]を選択します



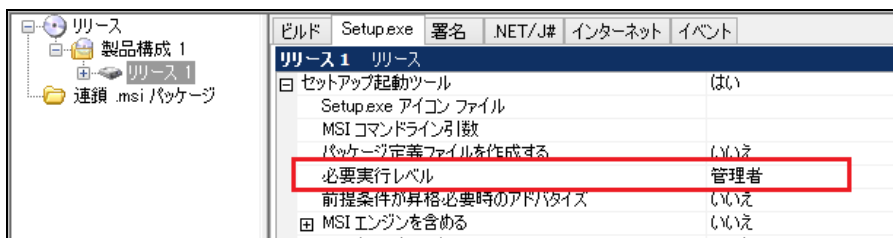
## 6. カスタムアクション名(WindowsFeatureEnable 等)を指定し、以下の設定を行います

設定プロパティ名	設定内容
作業ディレクトリ	SystemFolder
ファイル名とコマンドライン	cmd.exe /C "[SUPPORTDIR]¥WindowsFeatureEnable.bat"
スクリプト内実行	即時実行
インストール実行シーケンス	InstallFinalize の後
インストール実行条件	not Installed

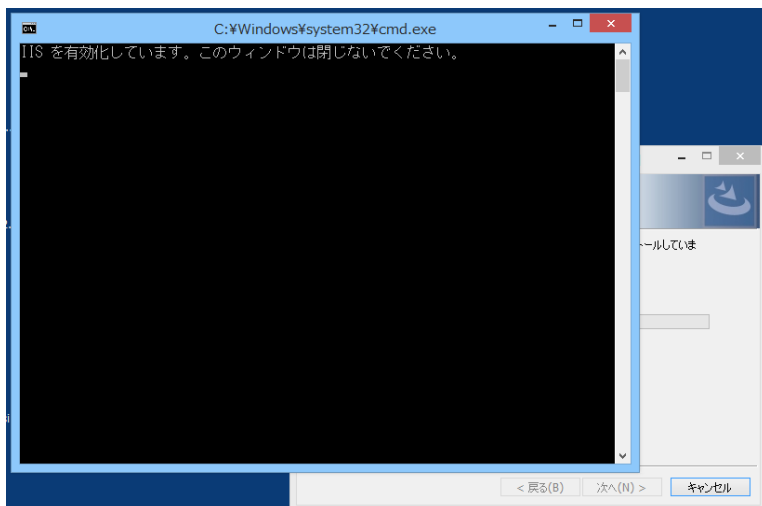


## 7. ビルドして、実行します

※コマンドの実行には管理者権限が必要なため、リリースの設定で[必要実行レベル]を[管理者]へ変更してください。



インストーラーの実行中に以下の画面が表示され、IIS の有効化処理が実行されます



## <InstallScript プロジェクトの場合>

### 1. バッチファイルを作成します

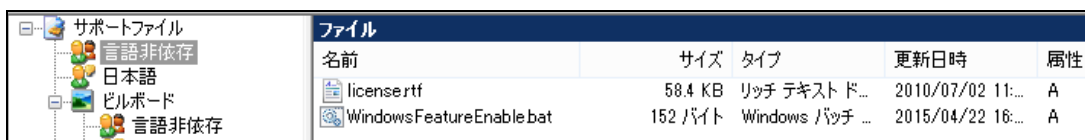
例: WindowsFeatureEnable.bat の内容

```
@echo off
echo IIS を有効化しています。このウィンドウは閉じないでください。
%windir%\system32\dism.exe /online /quiet /enable-feature /featurename:IIS-WebServerRole
```

### 2. 製品のプロジェクトを開き、[動作とロジック]—[サポート ファイル/ビルボード] ビューにて、[サポートファイル] のツリーを展開して、[言語非依存] を選択します

### 3. [ファイル]ペインを右クリックして、[ファイルの挿入]を選択します。ファイル選択のダイアログで、“WindowsFeatureEnable.bat”を選択します

### 4. “WindowsFeatureEnable.bat” がサポートファイルとしてインストーラーに追加されたことを確認します



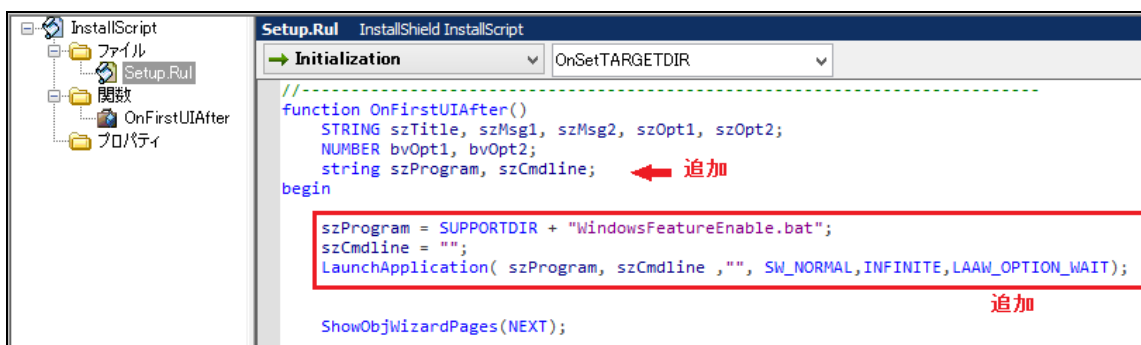
名前	サイズ	タイプ	更新日時	属性
license.rtf	58.4 KB	リッチ テキストド...	2010/07/02 11:...	A
WindowsFeatureEnable.bat	152 バイト	Windows バッチ ...	2015/04/22 16:...	A

### 5. [動作とロジック]—[InstallScript] ビューで、[Setup.rul] を表示します

### 6. [イベントカテゴリ] から「After Move Data」を選択し、[イベント] から「OnFirstUIAfter」をクリックします

### 7. Setup.rul に OnFirstUIAfter イベントが追加されるので、以下のスクリプトを追加します

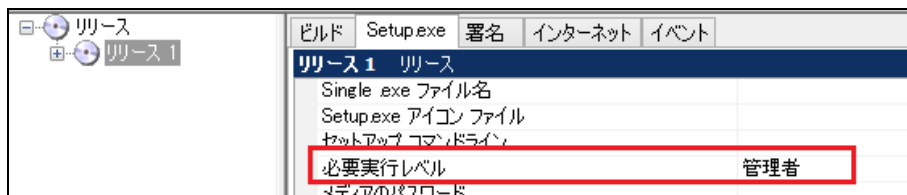
```
function OnFirstUIAfter()
:
string szProgram, szCmdline;
begin
szProgram = SUPPORTDIR + "WindowsFeatureEnable.bat";
szCmdline = "";
LaunchApplication( szProgram, szCmdline, "", SW_NORMAL,INFINITE,LAAW_OPTION_WAIT);
:
```



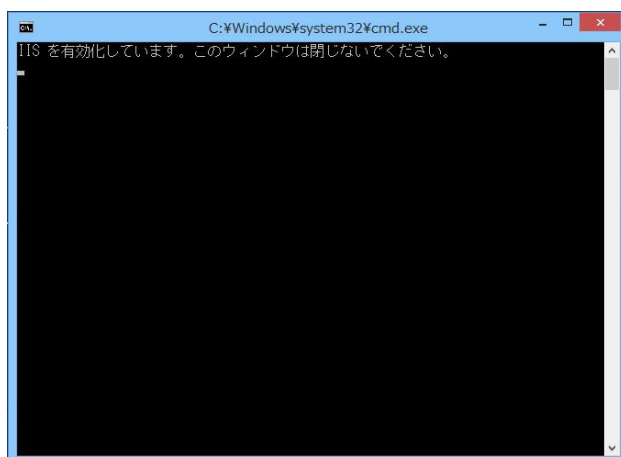
```
function OnFirstUIAfter()
:
string szProgram, szCmdline;
begin
szProgram = SUPPORTDIR + "WindowsFeatureEnable.bat";
szCmdline = "";
LaunchApplication( szProgram, szCmdline, "", SW_NORMAL,INFINITE,LAAW_OPTION_WAIT);
:
ShowObjWizardPages(NEXT);
```

## 8. ビルドして、実行します

※コマンドの実行には管理者権限が必要なため、リリースの設定で[必要実行レベル]を[管理者]へ変更してください。



インストーラーの実行中に以下の画面が表示され、IIS の有効化処理が実行されます



以上