

基本の MSI プロジェクトでカスタムダイアログを作成する

検証したバージョン: InstallShield 2010 Premier Edition

概要

InstallShield であらかじめ用意されているダイアログ以外に、カスタムダイアログを作成することができます。この記事では、基本のMSI プロジェクトでのカスタムダイアログの作成、シーケンスへのダイアログを挿入、コントロールイベントの使用方法などを紹介します。

■基本の MSI プロジェクトでカスタムダイアログを使用する

新しいダイアログを作成して、それをインストールで表示するには、以下のようなプロセスが必要となります。

- ・ダイアログの作成
- ・ダイアログのレイアウトの編集
- ・コントロールの動作を定義（表示する条件、イベントの設定など）
- ・ダイアログの表示

それぞれのプロセスについて、説明します。

■新しいダイアログの作成

まず初めに、以下の手順で新しいダイアログを作成します。

1. [ユーザー インターフェイス] -[ダイアログ]ビューを表示します。
2. [ダイアログ] エクスプローラで、[すべてのダイアログ] を右クリックしてから、[新しいダイアログ] をクリックします。ダイアログ ウィザードが起動し、新規ダイアログ作成手順を案内します。
3. [ダイアログテンプレート]画面で、作成したいダイアログを選択して、次へボタンをクリックします。
ここでは、内部ダイアログを選択します。

ダイアログテンプレート	説明
ブランク	ダイアログコントロールを含まない空白のダイアログ
外部	[ようこそ] ダイアログのフォーマットに似た、ダイアログの左側に大きなビットマップを持つダイアログ。ウィザードの始めと終わりに表示するダイアログです
内部	ダイアログの上部に小さなビットマップを持つダイアログ。通常、ウィザードの中間で使用されるダイアログです。

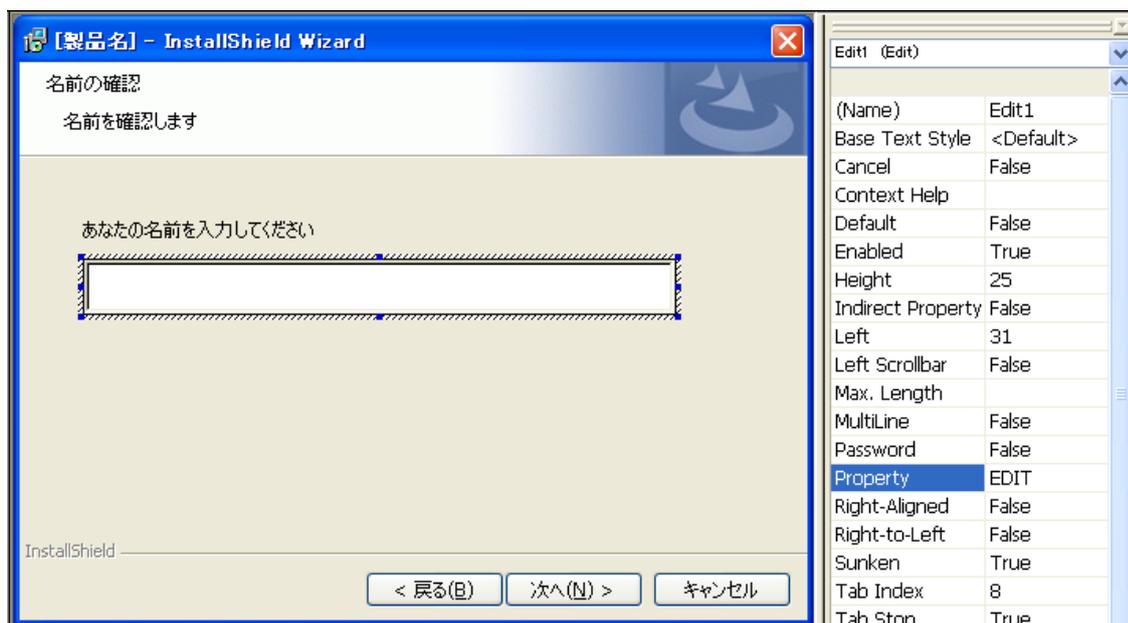
※その他のテンプレートについての説明は、製品ヘルプをご参照ください。

- [ユーザーインターフェイスシーケンス]パネルで、シーケンスを選択します。
通常ダイアログはインストールシーケンスに挿入しますので、[ユーザーインターフェイスシーケンス]を[シーケンス]に設定します。また、ダイアログを挿入する位置を選択します。ここでは、[InstallWelcome]を選び、次へボタンをクリックします。
- [ダイアログの位置と条件]パネルで、ダイアログを表示する位置を指定します。
ここでは、LicenseAgreement ダイアログ(使用許諾ダイアログ)の次に表示されるように、[Interior]を選択して、[下に移動]ボタンでLicenseAgreement の下に配置します
[条件]は[1]のまま、次へボタンをクリックします。
※条件が1のとき、常にこのダイアログが表示されます
- ウィザードの[完了]ボタンをクリックすると、新しいダイアログが作成されるので、任意のダイアログ名を設定します。ここでは「MyDialog」とします

■ダイアログのレイアウトの編集(コントロールの配置)

新しいダイアログが作成できたら、ダイアログのレイアウトを編集します。今回は、編集フィールドを作成して、ユーザに名前を入力してもらうようなダイアログレイアウトにします。

- 新規に作成したダイアログ MyDialog の[日本語]ノードを選択します
- ダイアログタイトル(DlgTitle)やダイアログの詳細(DlgDesc)を適宜変更してください
- ツールバーの **A** ボタン(テキスト領域)を選択し、ダイアログ上でドラッグします。テキスト領域が作成されたら、Text プロパティに「あなたの名前を入力してください」と入力します
- 次に、ツールバーの **labl** ボタン(編集フィールド)を選択し、ダイアログ上でドラッグします。コントロール プロパティ画面が表示されたら、任意のプロパティ名(ここでは EDIT)を入力します。実行時にユーザがテキストを入力すると、EDIT プロパティに入力したテキストが保存されます。



5. [動作とロジック]—[プロパティマネージャ]でEDIT を空欄(NULL)にします

名前 :EDIT

値 :空欄(NULL)

■コントロールの動作を定義 (表示する条件、イベントの設定など)、ダイアログの表示

ダイアログコントロールの動作を編集します。

今回は、VBScript のメッセージボックスを使用したカスタムアクションを、[次へ]ボタンが押されたタイミングで実行する動作を定義します。

編集フィールドに入力された値をメッセージボックスに表示し、入力された名前が正しいかどうかを確認して[はい]が押された場合に次のダイアログへ遷移するような動作を定義します。

1. 入力された名前を表示するVBScript のカスタムアクションを表示します。

[カスタムアクションとシーケンス]ビューでカスタムアクションアイコンを右クリックして、[新しい VBScript]—[カスタム アクションに保存]を選択します。カスタムアクション名を設定します。caCheckName

2. [スクリプト]タブで以下のコードを記述します。

```
-----
If MsgBox(Session.Property("EDIT") + "さんですね?", vbYesNo, "確認") = vbYes Then
    Session.Property("ANSWER")="YES"
Else
    Session.Property("ANSWER")="NO"
End if
-----
```

【Memo】

今回のサンプルでは、VBScript でプロパティの値を取得したり、設定したりするコードを使用しています。

```
Session.Property("EDIT")
```

EDIT というプロパティの値(編集フィールドに入力された値)を VBScript で取得します

```
Session.Property("ANSWER")="YES"
```

ANSWER というプロパティに VBScript から値(YES)を代入しています。VBScript の結果をプロパティの値に代入することで、Windows Installer 側で VBScript の結果に応じた処理ができるようになります。

3. MyDialog の[動作]ノードを表示します

次へボタンが押されたタイミングでカスタムアクションが起動するように、Next コントロールの[Events]タブで以下のように編集します

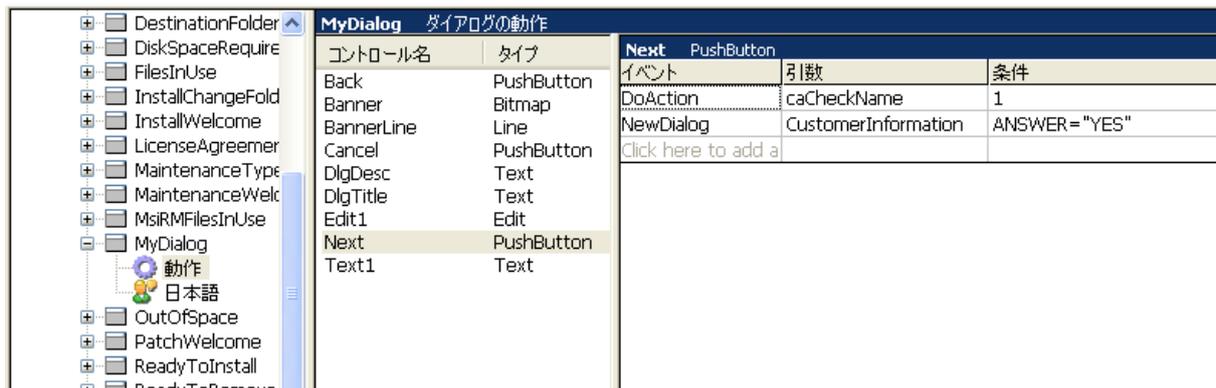
イベント	引数	条件
DoAction	caCheckName	1

DoAction イベントは、引数に指定されたカスタムアクションをダイログから起動するイベントです。

DoAction イベントの下に、以下のイベントを追加します。

イベント	引数	条件
NewDialog	CustomerInformation	ANSWER = "YES"

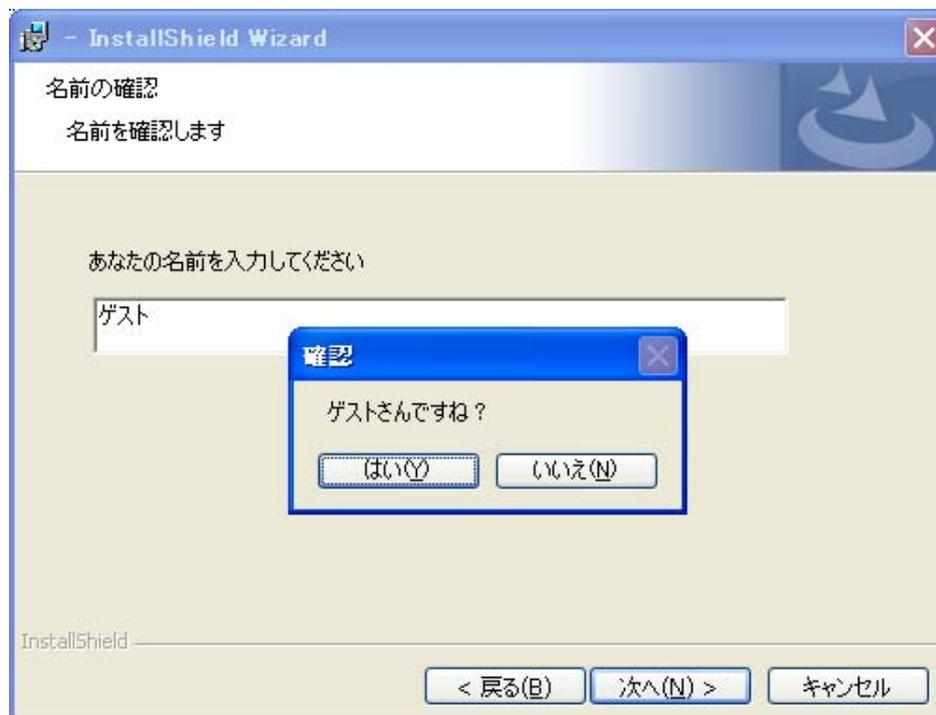
NewDialog イベントは、引数に指定されたダイアログを起動するイベントです。条件に ANSWER = "YES" と記述することで、caCheckName カスタムアクションでプロパティに YES が設定されたとき(メッセージボックスではないが選択されたとき)だけユーザ情報ダイアログを表示するような動作としています。



■インストーラの実行

ビルドしてインストーラを実行すると、[使用許諾契約]画面の次に、カスタムダイアログが表示されます。

編集フィールドに名前を入力して、次へボタンをクリックすると、以下の画面が表示されます。



[はい]をクリックすると、[ユーザ情報]のダイアログに遷移します。[いいえ]をクリックした場合には、ダイアログの遷移は行わず、名前の修正を行うことができます。