Dave's Development Blog

Software Development using Borland / Codegear / Embarcadero RAD Studio

OTA Template Wizard and Notifier Indexes…

By David | November 8, 2011 0 Comment

I found a problem with my implementation with wizards and notifiers in add-ins the other day when I kept getting a crash on recompiling a Package in the IDE. It transpired that the problem was that I did not remove a compiler notifier on closing down the package so the IDE tried to use the notifier when it was not present. The reason for all of this was that the index returned by **AddNotifier()** returned a zero.

Now somewhere in the dim and distant past I thought I read that zero meant that the notifier was not install. It would seem that this was wrong and zero is a valid index. Having a look through all **AddNotifier()** methods in the ToolsAPI.pas module shows in some places that an error is indicated by the index return being `< 0`. Thus I've re-coded the installation and removal of wizards and notifier as follows.

First change the default starting index of the wizards and notifier to -1 as in the below code (this is now done through a constant):

```
Const
  iWizardFailState = -1;

Var
  ...
  {$ENDIF}
  iWizardIndex           : Integer = iWizardFailState;
  {$IFDEF D0006}
  iAboutPluginIndex      : Integer = iWizardFailState;
  {$ENDIF}
  iKeyBindingIndex       : Integer = iWizardFailState;
  iIDENotfierIndex       : Integer = iWizardFailState;
  {$IFDEF D2010}
  iCompilerIndex         : Integer = iWizardFailState;
  {$ENDIF}
  {$IFDEF D0006}
  iEditorIndex           : Integer = iWizardFailState;
  {$ENDIF}
  iRepositoryWizardIndex : Integer = iWizardFailState;
```

The code for installing the wizards and notifier has not changed but is included here for clarity:

```
Function InitialiseWizard(WizardType : TWizardType) : TWizardTemplate;

Var
  Svcs : IOTAServices;

Begin
  ...
  // Create Keyboard Binding Interface
  iKeyBindingIndex := (BorlandIDEServices As IOTAKeyboardServices).AddKeyboardBinding(
    TKeybindingTemplate.Create);
  // Create IDE Notifier Interface
  iIDENotfierIndex := (BorlandIDEServices As IOTAServices).AddNotifier(
    TIDENotifierTemplate.Create);
  {$IFDEF D2010}
  // Create Compiler Notifier Interface
  iCompilerIndex := (BorlandIDEServices As IOTACompileServices).AddNotifier(
    TCompilerNotifier.Create);
  {$ENDIF}
  {$IFDEF D2005}
  // Create Editor Notifier Interface
  iEditorIndex := (BorlandIDEServices As IOTAEditorServices).AddNotifier(
    TEditorNotifier.Create);
  {$ENDIF}
  // Create Project Repository Interface
  iRepositoryWizardIndex := (BorlandIDEServices As IOTAWizardServices).AddWizard(
    TRepositoryWizardInterface.Create);
End;
```

The last area of change is in the **Finalization** section where instead of checking for zero we check for -1 as follows:

```
Initialization
  ...
Finalization
  // Remove Wizard Interface
  If iWizardIndex > iWizardFailState Then
    (BorlandIDEServices As IOTAWizardServices).RemoveWizard(iWizardIndex);
  {$IFDEF D2005}
  // Remove Aboutbox Plugin Interface
  If iAboutPluginIndex > iWizardFailState Then
    (BorlandIDEServices As IOTAAboutBoxServices).RemovePluginInfo(iAboutPluginIndex);
  {$ENDIF}
  // Remove Keyboard Binding Interface
  If iKeyBindingIndex > iWizardFailState Then
    (BorlandIDEServices As IOTAKeyboardServices).RemoveKeyboardBinding(iKeyBindingIndex);
  // Remove IDE Notifier Interface
  If iIDENotfierIndex > iWizardFailState Then
```

```
    (BorlandIDEServices As IOTAServices).RemoveNotifier(iIDENotfierIndex);
  {$IFDEF D2010}
  // Remove Compiler Notifier Interface
  If iCompilerIndex <> iWizardFailState Then
    (BorlandIDEServices As IOTACompileServices).RemoveNotifier(iCompilerIndex);
  {$ENDIF}
  {$IFDEF D2005}
  // Remove Editor Notifier Interface
  If iEditorIndex <> iWizardFailState Then
    (BorlandIDEServices As IOTAEditorServices).RemoveNotifier(iEditorIndex);
  {$ENDIF}
  // Remove Repository Wizard Interface
  If iRepositoryWizardIndex <> iWizardFailState Then
    (BorlandIDEServices As IOTAWizardServices).RemoveWizard(iRepositoryWizardIndex);
End.
```

This bug would only have shown up in very rare circumstances but hopefully we can all learn from it.

Dave 🙂

Category: Open Tools API Tags: Borland, BorlandIDEServices, CodeGear, Delphi, Embarcadero, Experts, IOTAWizard, OTA, RAD Studio