Updating the User Interface

原文<https://docs.microsoft.com/zh-cn/visualstudio/extensibility/updating-the-user-interface>

本文内容

See Also

After you implement a command, you can add code to update the user interface with the state of your new commands.

In a typical Win32 application, the command set can be continuously polled and the state of individual commands can be adjusted as the user views them. However, because the Visual Studio shell can host an unlimited number of VSPackages, extensive polling might decrease responsiveness, especially polling across interop assemblies between managed code and COM.

To update the UI

**1.** Perform one of the following steps:

• Call the [UpdateCommandUI](https://msdn.microsoft.com/en-us/library/bb143525(v=vs.140).aspx) method.

An [IVsUIShell](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.ivsuishell(v=vs.140).aspx) interface can be obtained from the [SVsUIShell](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.svsuishell(v=vs.140).aspx) service, as follows.

void UpdateUI(Microsoft.VisualStudio.Shell.ServiceProvider sp)

{

IVsUIShell vsShell = (IVsUIShell)sp.GetService(typeof(IVsUIShell));

if (vsShell != null)

{

int hr = vsShell.UpdateCommandUI(0);

Microsoft.VisualStudio.ErrorHandler.ThrowOnFailure(hr);

}

}

If the parameter of the [UpdateCommandUI](https://msdn.microsoft.com/en-us/library/bb143525(v=vs.140).aspx) is non-zero (TRUE), then the update is performed synchronously and immediately. We recommend that you pass zero (FALSE) for this parameter to help maintain good performance. If you want to avoid caching, apply the DontCache flag when you create the command in the .vsct file. Nevertheless, use the flag cautiously or performance might decrease. For more information about command flags, see the [Command Flag Element](https://docs.microsoft.com/zh-cn/visualstudio/extensibility/command-flag-element) documentation.

In VSPackages that host an ActiveX control by using the in-place activation model in a window, it might be more convenient to use the [UpdateUI](https://msdn.microsoft.com/en-us/library/bb133682(v=vs.140).aspx) method. The [UpdateCommandUI](https://msdn.microsoft.com/en-us/library/bb143525(v=vs.140).aspx) method in the [IVsUIShell](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.ivsuishell(v=vs.140).aspx) interface and the [UpdateUI](https://msdn.microsoft.com/en-us/library/bb133682(v=vs.140).aspx) method in the [IOleInPlaceComponentUIManager](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.ioleinplacecomponentuimanager(v=vs.140).aspx) interface are functionally equivalent. Both cause the environment to re-query the state of all commands. Typically, an update is not performed immediately. Instead, an update is delayed until idle time. The shell caches the command state to help maintain good performance. If you want to avoid caching, apply the DontCache flag when you create the command in the .vsct file. Nevertheless, use the flag cautiously because performance might decrease.

Notice that you can obtain the [IOleInPlaceComponentUIManager](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.ioleinplacecomponentuimanager(v=vs.140).aspx) interface by calling the QueryInterface method on an [IOleComponentUIManager](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.iolecomponentuimanager(v=vs.140).aspx) object or by obtaining the interface from the [SOleComponentUIManager](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.interop.solecomponentuimanager(v=vs.140).aspx) service.

See Also

[How VSPackages Add User Interface Elements](https://docs.microsoft.com/zh-cn/visualstudio/extensibility/internals/how-vspackages-add-user-interface-elements)

[Implementation](https://docs.microsoft.com/zh-cn/visualstudio/extensibility/internals/command-implementation)