How to Use a Modeless WinForm and Still Immediately Unload your .DLL

Many programmers create WinForms for user interaction with NXOpen custom automation routines. WinForms can be displayed as either modal or modeless dialogs. If you post your form using ShowDialog(), the resulting dialog is modal, and the user cannot perform any other actions within NX while the dialog is displayed. This is often inconvenient, as the user might need to move the model around or query something with the Information menu, for example. If you post the form with Show() instead, then the dialog is modeless. In this state, some interaction with NX is allowed, but it is not possible to use the unload option "Immediately", because doing so will cause the form to be dismissed as soon as it is posted.

During the program development and testing process, it saves time if the program can be unloaded immediately, so that the programmer can make changes to the code, re-build, and test again without taking the steps required to explicitly unload the .DLL.

Once the program has been deployed to your user community, you might want the program to unload immediately for a variety of reasons, such as to release any licenses the program was reserving, or perhaps to make a small change to the library, which you cannot do if a user has loaded the library and then left his session running. Whether he is still using the .DLL or not, if it remains loaded, the programmer cannot update it.

The solution to this quandary is to create a second thread within your program that will check to see whether the form has been dismissed. In general, we do not recommend using multiple threads with NXOpen code, as NXOpen is not considered to be "thread-safe". However this method is only making one call to NX code – the one to unload the library. The rest of the time it is just checking to see whether the form is still there, or counting the clock ticks. Note that the code shown below is actually only performing the check for the form dismissal every thirty seconds. This seems immediate enough for most occasions. You can tinker with this delay period by changing 30000 to some other value, or you can remove the call to Thread.Sleep() completely.

This method will allow you to use modeless dialogs, so that the user can interact with NX while the form is displayed, and still retain the desired behavior of unloading the .DLL when the program is finished. (The form code is omitted intentionally for brevity.)

```
Option Strict Off
Imports System
Imports System.Threading
Imports System.Reflection
Imports System.Windows.Forms
Imports NXOpen
Imports NXOpen.UF

Module unload_modeless_winform_immediately

    Dim myform As New Form1
    Dim s As Session = Session.GetSession()
    Dim ufs As UFSession = UFSession.GetUFSession()

    Public Sub Main()

        myform.Show()

        Dim checkThread As New Thread(New ThreadStart(AddressOf IsFormDismissed))
        checkThread.Start()

    End Sub

    Public Sub IsFormDismissed()

        Do
            If myform.IsDisposed() = True Then
                UnloadNXLibrary()
            End If
            Thread.Sleep(30000)
        Loop

    End Sub

    Sub UnloadNXLibrary()
        Dim runningProgram As Assembly = Assembly.GetExecutingAssembly()
        ufs.UF.UnloadLibrary(runningProgram.Location)
    End Sub

    Public Function GetUnloadOption(ByVal dummy As String) As Integer

        GetUnloadOption = NXOpen.Session.LibraryUnloadOption.Explicitly

    End Function

End Module
```

Steve Labout