How can I determine the reason why my window is closing?

devblogs.microsoft.com/oldnewthing/20190411-00

April 11, 2019

Raymond Chen

A customer wanted to know whether their MFC program can determine why their window is closing, similar to how WinForms does.

The source code for WinForms is online. You can see how they do it, and then translate that to MFC.

Many of the `CloseReason` values refer to actions that occurred within WinForms itself, so those are naturally generated by WinForms. Three of the reasons are external to WinForms.

- `UserClosing`: This is generated in response to the `WM_SYSCOMMAND` when the code is `SC_CLOSE`. This happens when the user closes the window by clicking the × in the upper right corner, or by double-clicking the system icon, or by selecting *Close* from the system menu.
- `WindowsShutDown`: This is generated in response to the `WM_QUERYENDSESSION` and `WM_ENDSESSION` messages.
- `TaskManagerClosing`: This is generated in response to the `WM_CLOSE` message, provided it wasn't already set by someone else with better information.

The "provided it wasn't already set by someone else with better information" is important, because many of the window closing scenarios lead to `WM_CLOSE`. For example, the default handling for the `SC_CLOSE` system menu command is to send the `WM_CLOSE` message, so you will see the `SC_CLOSE` first, followed by the `WM_CLOSE` message.

Note that `TaskManagerClosing` is inferred by the fact a `WM_CLOSE` message arrives without any of the known preliminaries. While it's true that Task Manager uses the `WM_CLOSE` message to encourage an app to exit, it's not the only program that does it.

A better name might be `External` or `Programmatic`.

Raymond Chen

Follow