

# Sometimes a function returns NULL because NULL really is the answer

[devblogs.microsoft.com/oldnewthing/20151105-00](http://devblogs.microsoft.com/oldnewthing/20151105-00)

November 5, 2015



Raymond Chen

A customer reported that sometimes the `GetGuiThreadInfo` function returned a valid window handle, but sometimes it returned all `NULL`s.

```
DWORD dwThreadId = GetWindowThreadProcessId(hwnd, NULL);
GUI_THREAD_INFO guiThreadInfo;
guiThreadInfo.cbSize = sizeof(GUI_THREAD_INFO);
if (GetGuiThreadInfo(dwThreadId, &guiThreadInfo)) {
    HWND hwndActive = guiThreadInfo.hwndActive;
    ...
}
```

Most of the time, the call to `GetGuiThreadInfo` succeeds and obtains `hwndActive` successfully. But sometimes, `GetGuiThreadInfo` succeeds, but `guiThreadInfo.hwndActive` is `NULL`. In fact, aside from the `cbSize`, all the members of the `guiThreadInfo` are `NULL` or zero. “Under what circumstances will `GetGuiThreadInfo` succeed but return no data?”

I suggested that they start with the obvious: Is it possible that the function is correct and the thread has no active window?

The customer confessed that they were too quick to assume that there was a problem with the `GetGuiThreadInfo` function just because it reported no data. It turns out that, in fact, there was no data to report.

The root cause was that another thread in their program called `SetWindowPos` and didn’t pass the `SWP_NOACTIVATE` flag. As a result, that thread stole activation from the first thread, so when they got around to asking the first thread, “Tell me about your active window, your focus window, your caret window, and your caret location,” it replied, “I don’t have any of those things!”

Raymond Chen

**Follow**

