

# Pumping messages while waiting for a period of time

 [devblogs.microsoft.com/oldnewthing/20060126-00](http://devblogs.microsoft.com/oldnewthing/20060126-00)

January 26, 2006



Raymond Chen

We can use the `MsgWaitForMultipleObjects` function (or its superset `MsgWaitForMultipleObjectsEx`) to carry out a non-polling “sleep while processing messages”.

```
#define MSGF_SLEEPMSG 0x5300
BOOL SleepMsg(DWORD dwTimeout)
{
    DWORD dwStart = GetTickCount();
    DWORD dwElapsed;
    while ((dwElapsed = GetTickCount() - dwStart) < dwTimeout) {
        DWORD dwStatus = MsgWaitForMultipleObjectsEx(0, NULL,
            dwTimeout - dwElapsed, QS_ALLINPUT,
            MWFO_WAITANY | MWMO_INPUTAVAILABLE);
        if (dwStatus == WAIT_OBJECT_0) {
            MSG msg;
            while (PeekMessage(&msg, NULL, 0, 0, PM_REMOVE)) {
                if (msg.message == WM_QUIT) {
                    PostQuitMessage((int)msg.wParam);
                    return FALSE; // abandoned due to WM_QUIT
                }
                if (!CallMsgFilter(&msg, MSGF_SLEEPMSG)) {
                    TranslateMessage(&msg);
                    DispatchMessage(&msg);
                }
            }
        }
    }
    return TRUE; // timed out
}
```

This function pumps messages for up to `dwTimeout` milliseconds. The kernel of the idea is merely to use the `MsgWaitForMultipleObjects/Ex` function as a surrogate for `WaitMessageTimeout`, pumping messages until the cumulative timeout has been reached. There are a lot of small details to pay heed to, however. I’ve linked them to earlier postings that discuss the specific issues, if you need a refresher. The `CallMsgFilter` you might find gratuitous, but you’ll change your mind when you realize that users might press a keyboard

accelerator while you're sleeping, and you presumably want it to go through somebody's `TranslateAccelerator`. The message filter lets you hook into the modal loop and do your accelerator translation.

Extending this function to “wait on a set of handles up to a specified amount of time, while pumping messages” is left as an exercise. (You can do it without changing very many lines of code.)

[Call the right function. -2pm]

Raymond Chen

**Follow**

