When you set a 100% CPU program to real-time priority, you get what you asked for

devblogs.microsoft.com/oldnewthing/20100610-00

June 10, 2010

Raymond Chen

Real-time priority is really dangerous. It's higher priority than nearly everything else. It's higher priority than mouse input, keyboard input, and the disk cache. If you foolishly set the priority class of a CPU-intensive program to real-time, it will suck up your entire processor, leaving no cycles for anything else. In particular, since not even input runs at real-time priority, you can't stop it via any interactive means, because the thread that manages input can't even run to process your input. Mind you, even if the input thread did run at real-time priority, that wouldn't really help you any. Sure, it could receive your input and distribute it to the appropriate application queues, but those applications are themselves not running with real-time priority, so all that happens is that your input gets quickly transferred to the input queues, where it then sits waiting for the applications to process them (which will never happen since the applications are not running with high enough priority). One might argue that Task Manager should run with real-time priority, so it can extricate you from this situation, but that won't help, and it would be wrong. It won't help because you first need to be able to launch Task Manager (or switch to it, if you were prescient enough to have it already running), and none of the ways of launching Task Manager run with real-time priority. (Because *nothing* in the user interface runs with real-time priority.) Second, even if there were a special code path that enabled you to launch Task Manager at real-time priority, it would be wrong, because the fact that Task Manager is running with real-time priority means that it is now stealing CPU cycles from that other process which set itself to real-time priority, which defeats the purpose of that process setting itself to real-time priority in the first place: It set itself to real-time priority because it didn't want anybody stealing CPU time from it! What could be done is to have Task Manager display an extra warning dialog when somebody uses it to change a process's priority. The warning dialog would say something like "Fiddling with priority can result in you being totally screwed. Are you sure you want to take this risk?" (Oh wait, *that dialog box already exists*.) Our friend Igor probably clicked right on past that warning dialog, because he's thinking, "Of course I want to set it to real-time priority, you stupid program, that's why I clicked *Real-Time Priority*! This is another in a long string of examples of how Windows is one of those coddling operating systems that gets in the way of advanced users like me, throwing up frustrating obstacles which prevent me from getting things done."

And then he shoots himself in the foot and asks to be coddled a little bit.