Operating Systems Programming Assignment Unit 5

Anonymous Student

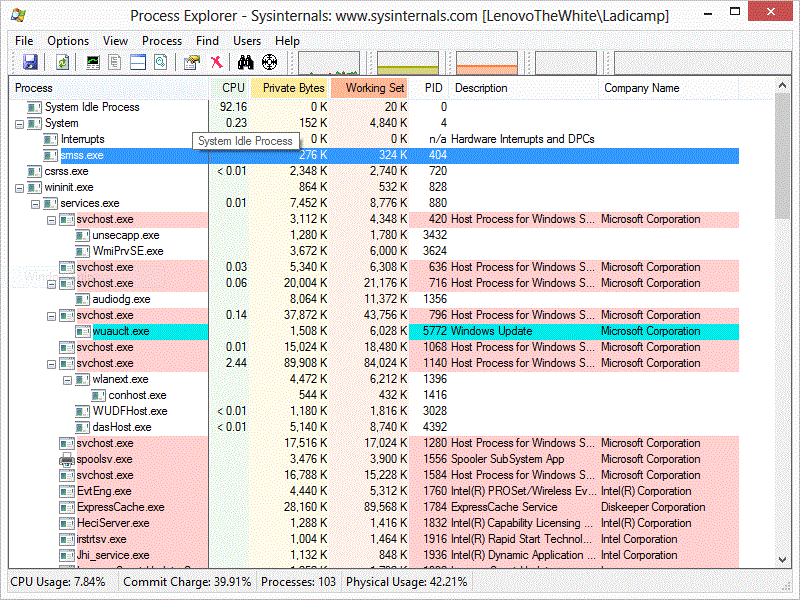
University of the People

Operating Systems Programming Assignment Unit 5

This report discusses the use of Process Explorer, a tool which can provide realtime information about a system’s running processes and their memory usage.

**1. Process Explorer Startup Screen**

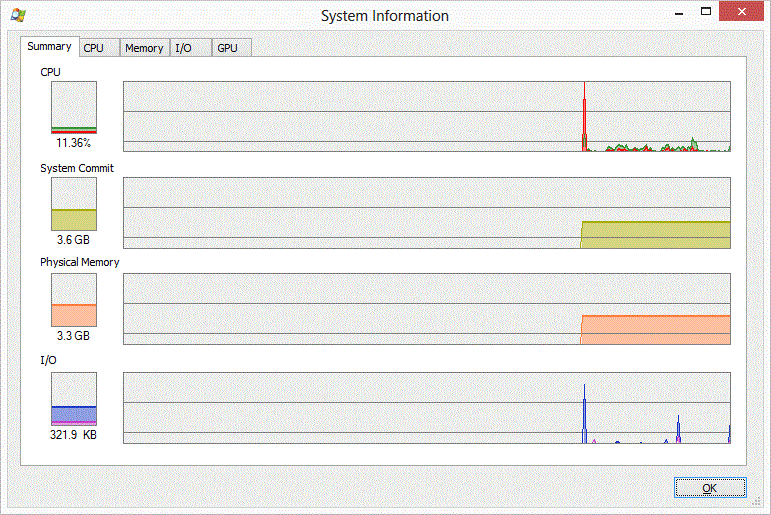
Below is a screen capture of the Process Explorer window when it first starts. Its startup screen shows a list of all of the processes being run on the computer in real time. In addition to



listing the processes, it also provides details about the process like the percentage of CPU being utilized, the address space dedicated to the process in bytes, the bytes of working memory, the process ID, a brief description of the process, and the company name it belongs to.

**2. System Information Window**

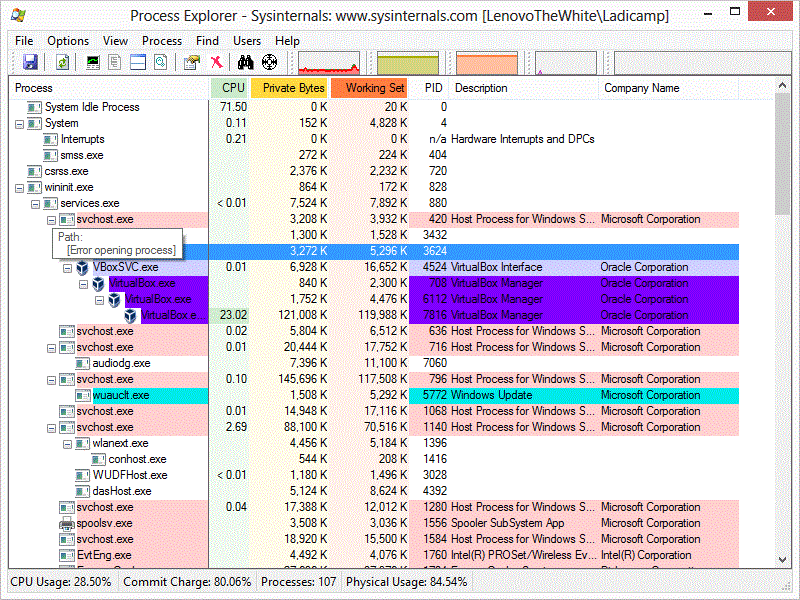
From the startup screen, a user can see detailed information about the memory activity of the entire system by opening the System Information window (by typing Ctrl-I). Specifically, the user can learn about kernel memory, and physical memory, and I/O processes. Here is a screen capture of the System Information window, which shows that 11.36% of the



CPU is being utilized (top left). Next down, the Kernel (or System Commit) is using 3.6 GB of memory. Following that, the Physical Memory being used is 3.3 GB. Finally, the I/O memory usage is 321.9 KB. The graphs to the right of each of these shows real-time changes in these values.

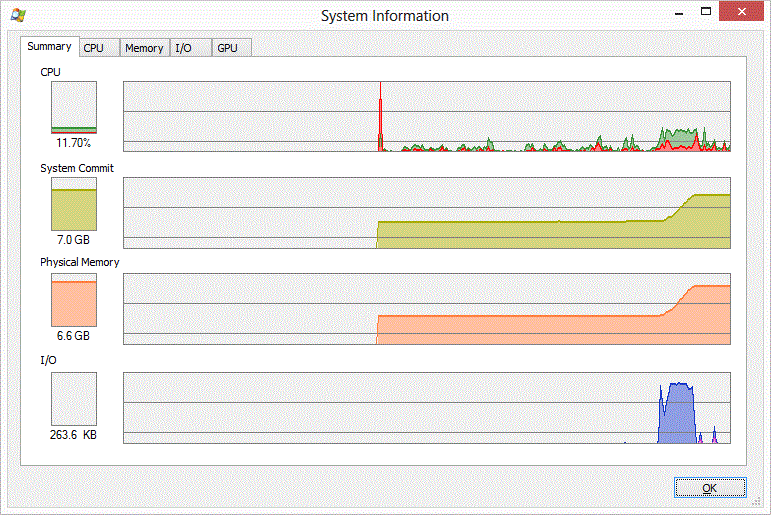
**3. Starting a New Program, 1 Minute of Runtime, and Termination**

When a user starts a new program, the Process Explorer screens change accordingly. First, the startup screen changes to show that a new process has been added to the list of processes. It was recommended for us to use Excel for this demonstration, but I do not own Excel. Therefore I opened Oracle VM Virtual Box instead. In the screen capture below, the blue process in the middle of the screen shows that four VirtualBox/Oracle Corporation

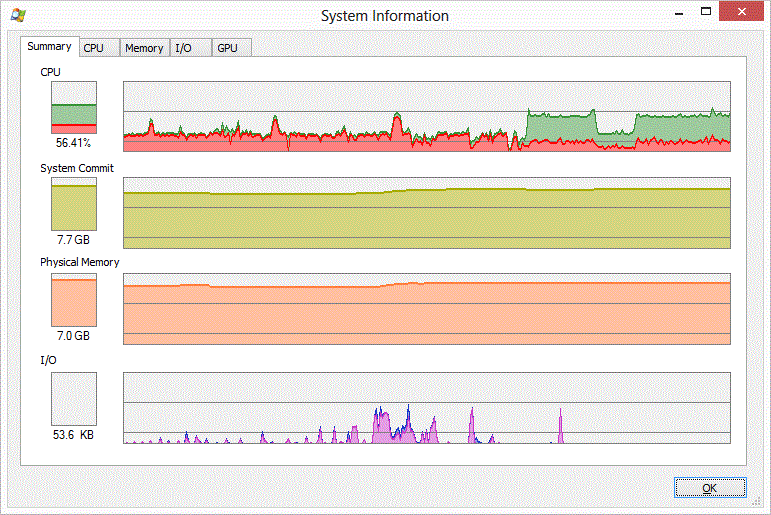


processes are running. In addition to this, the system information window changes (see the 3 images on the following page). When the program is first started, it shows a steep rise in activity on all measurements at the point where the new application was opened. The two measurements that are the most telling are the Kernel usage (System Commit) and the Physical Memory usage, both of which nearly doubled in magnitude and remain high even after a minute of the program running. Both of these continue to remain at roughly the same level until the program is killed. As for CPU usage and I/O usage, there was a spike of activity, but these have returned to previous levels after the application was started and running for about a minute.

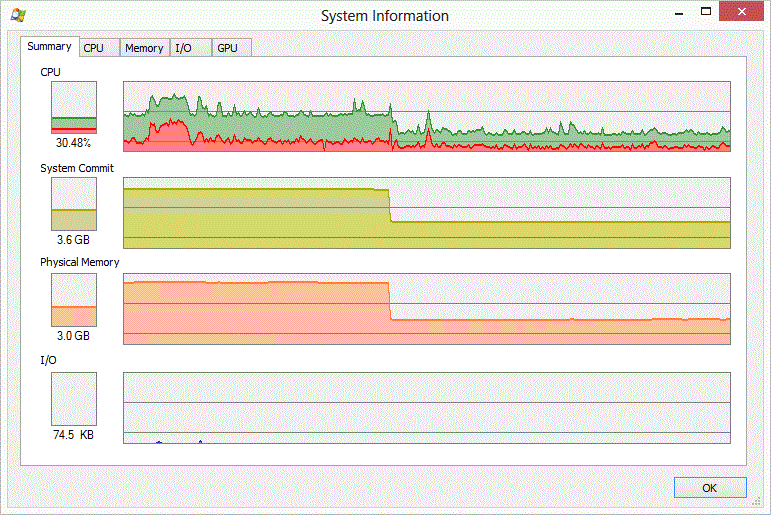
Starting a New Program



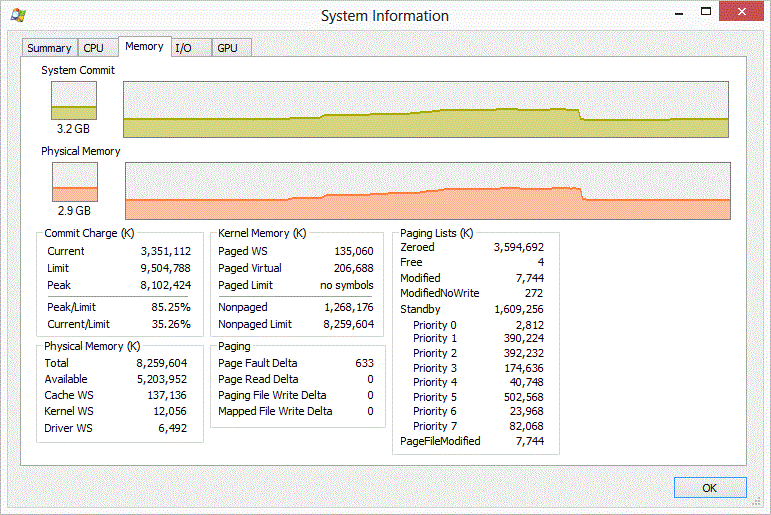
Program After 1 Minute (or More) of Runtime



Terminating a Program



**3. Paging**

In order to look at paging information, the Memory tab of the System Information window needs to be active. Here is what that tab looks like. This provides a glimpse of paging 

in real-time. My understanding of paging is limited to scanning ahead a few chapters in our textbook, which we are not yet responsible for this week (Arpaci-Dusseau & Arpaci-Dusseau, 2012). It seems that paging is used to extend virtual memory by storing background process information into a file on the hard drive instead of keeping it active in main memory. As a new memory-costly program is started, the paging activity should increase. Conversely when a program is terminated, paging activity should decrease.

**Conclusion**

This week, we were asked to use Process Explorer to get a better understanding of what the computer system is doing as programs are starting, running, and terminating. Using this utility, we can determine how much of the CPU is being utilized in general. Additionally, we can determine, how much the kernel memory is being used, physical memory and look at paging as well. When a program first starts there is a spike in activity on all measures. As the program continues to run, the kernel and physical memories continue to be in use and elevated, whereas the CPU usage and I/O usage level out. It is only when a program is terminated that memory is deallocated and the kernel and physical memory drops back to a level consistent with what it was prior to the program starting. Additionally, paging increases when new programs are started because other programs in the background may need to keep process information stored in files rather than in main memory.

References

Arpaci-Dusseau, R. & Arpaci-Dusseau, A. (2012). *Operating Systems: Three Easy Pieces. Madison, WI: University of Wisconsin-Madison.* Retrieved from <http://pages.cs.wisc.edu/~remzi/OSTEP//>

Techyv (Sep 10, 2012). How to use Process Explorer to see what processes are doing. *YouTube*. Retrieved from <https://www.youtube.com/watch?v=bPI0xE8F3qc>

“Using Process Explorer.” (n.d.). *Kansas State University.* Retrieved from <https://www.k-state.edu/its/security/training/2009-4-9/presentations/handouts/Process_Explorer_Tutorial_Handout.pdf>