

Re: few questions about how linux work

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.development.system/2003-08/0127.html>

From: udayan (udayan_singh_at_lko.tcs.co.in)

Date: 08/06/03

Date: 6 Aug 2003 07:05:27 -0700

"Markus Falkensteiner" <falkmREMOVE_THIS@gmx.net> wrote in message news:<[B95Ya.14\\$wm6.600175@news.salzburg-online.at](mailto:B95Ya.14$wm6.600175@news.salzburg-online.at)>...

- > *second question refer to the scheduler:*
- > *the scheduler is the part which gives every process (which would like like*
- > *to get CPU)*
- > *a timeslice. this is as much time as the process wants or maximum 10ms. then*
- > *an timer-interrpt occurs*
- > *and the scheduler switch to next process.*
- > *i've heart that kernel-code can not loose time, means kernelcode can't be*
- > *interrupted. what is*
- > *when a kernel function needs more that 10ms (however) and the next*
- > *timer-interrupt occurs.*
- > *will this IRQ be ignored. if yes how does this works. or will the interrupt*
- > *(any interrupt) processed later when kernelcode is finished??*
- >

well i have some knowladge with respect to the theoretical part of some operating systems. Out there it has two things (and many more) related to each process for using the CPU –

1. Time slice
2. Priority.

Now lets say we have a system on which we have 4 processes running of which one of them is having priority 1 (consider it to be kernel process and call it process 1, higher priority) and other 3 have priority 2 (user processes from 2–4, lower priority) .

Here if we consider both the above mentioned features being implemented, then what happens is that when process 1 is idle then other 3 processes run in round robin manner. with a time slice of t ms. But when process 1 (kernel process)comes into picture then it'll get the complete time till it completes off with its CPU utilization time. So you can see priority plays a role in it.

Havent worked on the process scheduling part on linux so cannot explicitly tell you what happens out here, but i suppose it shuold be somewhat related to what i said.. but someone else can can tell you

comp.os.linux.development.system: Re: few questions about how linux work
elaborately about linux kernel.

one example of this category is Compatible Time Sharing Systems.