

Operating Systems

COMS W4118

Lecture 24

Alexander Roth

2015 – 04 – 23

1 Scheduling II

1.1 Motivation

- There is no one-size-fits-all scheduling algorithm.
- The Completely Fair Scheduler is a complicated beast.
- Many real scheduling algorithms combine many elements of the simple algorithms that we see.

1.2 Combining Scheduling Algorithms

- The conceptual image is there are a list of running processes.
- The kernel picks one process to run next.
- The real picture is more complicated.
- There are multi-level queues.
- We have multiple queues because most systems have multiple CPUs.
- Each queue can have it's own scheduling algorithm.
- Foreground processes should have higher priority than the background processes.

1.3 Global Queue of Processes

- There is one ready queue shared across all CPUs

1.4 Per-CPU Queue Of Processes

- We statically partition processes to the CPU.
- Easy to implement, but there is the issue of a load imbalance.

1.5 Hybrid Approach

- Use a hybrid approach. Have a global queue and a per-CPU queue.
- Ability to move processes around.

1.6 SMP: “Gang” Scheduling

- If you have multiple squads that work together, producer-consumer model.
- These systems will work best if everything is working at the same time.
- The operating system will try to schedule the system together as much as possible.

1.7 Real-Time Scheduling

- Linux supports Real-Time Scheduling
- There are two kinds of real-time scheduling.
 1. Hard Real-Time, and
 2. Soft Real-Time
- Soft Real-Time: If you have a task that is designated as real-time, it will run and preempt any normal processes.
- Hard Real-Time is difficult to implement because of all the processes that are running.

1.8 Linux Scheduling Overview

- There are multiple levels of queues with priority
- There are hundreds of levels of priority.
- These are hard priority levels, so a higher priority preempts the lower level.
- 0 is the normal priority.
- There are sublevels of priorities in a hard level.
- Linux has 100 static priority levels.
- Nice values are the priority values within a level.

1.9 Completely Fair Scheduler (CFS)

- Calculates schedule latency: the amount of time it takes for each process to run at least once.
- Everyone has to run the schedule latency.
- CFS is not completely fair if there are more processes than you can fit in the latency period in a reasonable fashion.